



Blender wiki PDF Manual conversion by Marco Ardito

Details, info, download: <http://amrc.altervista.org>

Updated: 10/11/2014 from:

<http://wiki.blender.org/index.php/Doc:ET/2.6/Manual>



Table of Content

Table of Content	2
Mis on Blender?	41
Blenderi paigaldamine	41
Alustamine	41
Kasutajaliides	41
Peamised funktsioonid	41
Eelistuste seadistamine	41
Sinu esimene animatsioon	41
Andmesüsteem ja failid	41
Tegevused failidega	41
Blenderi andmesüsteem	41
Interaktsioonid 3D-ruumis	42
Sissejuhatus	42
3D-ruumis navigeerimine	42
Teisendused	42
Teisenduste piiramine	42
Skitseerimine 3D-ruumis	42
Modelleerimine	42
Sissejuhatus	42
Objektid	42
Võre objektid	42
Kõverad	43
Pinnad	43
Tekstiobjektid	43
Metaobjektid	43
Tühiobjektid	43
Grupiobjektid	43
Skriptid	43
Töötlemine ja muutamine	43
Sissejuhatus	43
Muuda	43
Moodusta	43
Moonuta	43
Simuleeri	43
Valgustamine	44
Sissejuhatus	44
Valgused	44
Varjud	44
Valgustid	44
Valgusvihud	44
Keskonna valgustus ja varjutamine	44
Materjalid	44
Sissejuhatus	44
Kasutamine	44
Omadused	44
Sõlmedest materjalid	45
Tippude maalimine	45
Halomaterjalid	45
Tekstuudid	45
Sissejuhatus	45
Tekstuuri tüübid	45
Laotamine	45
Mõju	45
Maailm ja keskkonna efektid	45
Maailm	45
Keskonna efektid	45
Taageldamine	45
Sissejuhatus	46
Skeletid	46
Nahaga katmine	46
Poseerimine	46

Piirajad	46
Sissejuhatus	46
Teisendamise piirajad	46
Järgimise piirajad	46
Suhte piirajad	46
Animeerimine	46
Sissejuhatus	46
Võtmekaadrid	46
Animatsiooniredaktorid	47
Animatsioonitehnikad	47
Füüsikaline simulatsioon	47
Sissejuhatus	47
Dünaamika	47
Osakesed	47
Pehme keha	47
Riie	47
Vedelikud	47
Dünaamiline maalimine	47
Mängumootori kasutamine	47
Jõudlus	47
Blenderi sisese mootoriga renderdamine	47
Sissejuhatus	47
Kaamerad	48
Renderdamine	48
Renderduse tulemus	48
Efektid ja järeltöötlus	48
Renderdusjõudlus	48
Välised renderdusmootorid	48
Renderdusmootoriga Cycles renderdamine	48
Sissejuhatus	48
Õppetükid	48
Mootori seletus	48
Sõlmedega komposiitimine	48
Komposiitimissõlmed	48
Komposiitimissõlmede tüübid	48
Montaaž	49
Sissejuhatus	49
Monteerija	49
Efektid	49
Heli	49
Blenderi laiendamine	49
Sissejuhatus	49
Pythoniga skriptimine	49
Pythoni skriptid	49
Mängumootor	49
Sissejuhatus	49
Loogika redaktor	49
Sensorid	49
Kontrollerid	49
Käitured	49
Kaamerad	50
Füüsika	50
Pythoni API	50
Erinevad allikad	50
Mis on Blender?	51
Video: Blender 1.60 kuni 2.50	52
Tähtsamad verstapostid	52
Eesti kogukond	55
Rahvusvaheline kogukond	55
IRC jututoad	55
Kes Blenderit kasutavad?	55
Sihtrühm	56
Arvutigraafika ja Blenderi õppimine	56
Rakenduse paigaldamine ja allalaadimine	58

Toetatud riistvara	58
Arendusplatvormid	58
Lähtekoodi kompileerimine	59
Pluginate kompileerimine	59
Laadi alla	61
Versioon	61
Paigaldamine	61
Tervitustekst	61
Litsentsitingimused	61
Paigaldusvalikud	61
Programmivalikud	61
Asukoht	62
Paigaldamine	62
Paigaldatud	62
Teisaldatav paigaldus	63
Distroga kaasatulev Blender	64
Kiire install	64
Põhjalikumad juhised	64
Kuidas KDE all Blenderi ikoon tekitada	64
Kuidas Blenderi ikooni GNOME'i all lisada	64
Ettevalmistused	70
Esmane allalaadimine ja paigaldus	70
Blenderi kompileerimine	70
Lae lähtetekst ja teegid	70
Kohanda seadistuste fail	70
Blender.app ehitamine	70
Puhastamine	70
FreeBSD	72
Teised ametlikult mittetoetatud operatsioonisüsteemid	72
MorphOS	72
Enda kataloogistruktuuri määramine	73
Selgitused kataloogidele	73
Internetiühendus	74
Ettevaatust kahtlastest allikatest .blend-faile või tööriistu laadides!	75
Kaitse	75
Salvesta oma töö regulaarselt	76
Kasutajaliidese põhialused	76
3 reeglit	77
Võimas kasutajaliides	77
Ülevaade	77
Selles käsiraamatus kasutatavad tähistused	79
Üldine kasutus	79
Hiirenupu emuleerimine	79
Numbriklahvide emulatsioon	79
Vaikestseen	80
Infoaken (Info Window) (või peamenüü)	80
3D-aken (3D Window)	81
3D-akna päis	81
Nuppude (omaduste) akna päis ('Buttons/Properties)	81
Ülevaateaken (Outliner Window)	81
Ajatelg (Timeline Window)	81
Akna maksimeerimine	82
Akna jagamine	82
Kahe paani liitmine	82
Akna suuruse muutmine	82
Sisu vahetamine	82
Uue akna avamine	82
Päise peitmine	84
Päise nähtavaks tegemine	84
Päise asend	84
Aknaatüübi nupp	84
Menüüd ja nupud	84
Windows 2000/XP/Vista/7	85
Linux	85
Tekstikonsooli teadaanded ja veateated	86
Tavalisemad teated	86
Vaata ka	87

Ekraanide seadistamine	88
Uue ekraani lisamine	88
Ekraani kustutamine	88
Ekraani ümberkorraldamine	88
Vaikeseadete üle kirjutamine	89
Täiendavad paigutused	89
Stseeni seadistamine	90
Uue stseeni lisamine	90
Lühike näide	90
Stseeni kustutamine	91
Toimingunupp	95
Lülisnupud	95
Valikunupud	95
Numbrinupp	95
Menüünupud	95
Värvivalija	96
Peidetud nupud	97
Mis on renderdamine?	98
Pildi renderdamine	98
Pildilahutuse määramine	98
Väljundi formaat ja väljundfail	98
Pildi salvestamine	98
Animatsiooni renderdamine	98
Ekraanipildid	99
Ekraanilindistus	99
Ekraanipilt operatsioonisüsteemi vahenditega	99
Ekraanipilt Windowsis	99
Ekraanipilt Mac OsX's	99
Ekraanipilt GNU/Linuxis	99
Kiirklahv	100
Menüüst	100
Tulemus	100
Blenderi standardse vaigestseeni taastamine	100
Tühistamine (Undo)	101
Ennistamine (Redo)	101
Tühistamise ajalugu (Undo history)	101
Salvestamine ja automaatne salvestamine	101
Automaatsete varukoopiate taastamine	102
Teised sätted	102
Veebipääs	103
Abimenüü (Help)	103
Veebipõhised abivõimalused	103
Sissehitatud abivõimalused	103
Seadistamine	104
Seadistuste salvestamine	104
Sissehitatud seadete taastamine	104
Omaduste aken (Properties)	105
Vaate muutmine (View Manipulation)	105
2D-vaateaknad (2D Viewports)	105
Manipulaator (Manipulator)	105
Menüüd (Menus)	105
New objects (uued objektid)	106
Undo (tühistamine)	106
Grease Pencil (rasvakriit)	106
Playback (taasesitus)	106
Keyframing (võtmekaadrid)	106
Transform (teisendus)	107
Duplicate Data (andmete duplitseerimine)	107
Häälestuste lisamine ja kustutamine	108
Häälestuste valimine	108
Vaikimisi häälestuse määramine	108
Klaviatuuri seadete eksport ja import	108
Mouse (hiir)	108
Numbrivälja emulatsioon	108
Vaate muutmine	108

Kiirklahvide redaktor	109
Teemade muutmine Pythoni abil	111
File Paths (failide asukohad)	118
Save & Load (salvestamine ja lugemine)	118
Auto Save (automaatsalvestus)	118
General (üldine)	120
Sound (heli)	120
Screencast (ekraanivideo)	120
Open GL	120
Sequencer (monteerija)	121
Solid OpenGL lights (OpenGL-i valgusallikad)	121
Miscellaneous (varia)	121
Soojendus	122
Keha ehitamine	123
Peeglis (Mirror) modelleerimine	125
Käed ja jalad	126
Tühistamine ja uuesti tegemine (Undo/Redo)	126
Kokkulangevad tipud	126
Pea	127
Pinnatükeldaja (Subdivision Surface / Subsurf)	127
Piiratud skaleerimine	128
Vaatame, milline Gus välja näeb	129
Kaamera seadistamine	130
Maapind	130
Valgusallikad	131
Renderdamine	132
Töö salvestamine	133
Materjalid ja tekstuurid	133
Silmad ja detailid	136
Koopia peegeldamine üle kursori	136
Suu	137
Silmade materjal	138
Renderdamine	138
Salvestamine	139
Taageldamine (Rigging)	140
Katmine	142
Tipugrupid	144
Poseerimine	144
Algne asukoht	145
Pöörkinemaatika	145
Edaspidine kinemaatika	145
Gus kõnnib!	147
Kirjeldus	149
Failisrivija kasutamine ja kataloogides navigeerimine	149
Kõrvalpaneel	149
Süsteem (System)	149
Järjehoidjad (Bookmarks)	149
Hiljutised (Recent)	149
Avamise valikud	149
Päise paneel	149
Muud failiavamise võimalused	150
Kirjeldus	151
Salvestamine	151
Salvestamise valikud	151
Failide pakkimine (Compress Files)	151
Säilita sõltuvused (Remap Relative)	151
Salvesta koopia (Save Copy)	151
Nõuanne failide nummerdamiseks	151
Võre	152
Import	152
Eksport	152
Valgus	152
Import	152
Eksport	152
Materjal & efektid	152
Sõlmed	152
Olulised asjad, mida meeles pidada	152
Ülevaade ja OOPS-skeem	153
Kasutajad (jagamine)	153

Võltskasutaja	153
Objektide kopeerimine ja linkimine stseenide vahel	153
Failidevaheline lisamine või linkimine	153
Proksiobjektid	153
Andmete kokku ja lahti pakkimine	153
Andmete lahti pakkimine	154
Ajatelje planeerimine	155
Kasutajaliidese laadimine menüü "File" → "Open" kaudu	156
Stseenidega töötamine	156
Stseeni lisamine	156
Stseeni nimetamine	156
Stseeni linkimine	156
Stseeni eemaldamine failist	157
Kirjeldus	158
Ülevaade	158
Ülevaateakna tüübi valimine	158
Ülevaate kasutamine	159
Valimine ja aktiveerimine	159
Objekti tasemel keeldude muutmine	160
Otsimine	162
Ülevaate nimekirja filtreerimine	162
Näide	162
Üldine töömeetod	164
Kataloogide ja failide organiseerimine	164
Teegi objektide avatud projekti lisamine	164
Objektide taaskasutus (Võred, Kõverad, Kaamera, Valgused...)	164
Materjalide/tekstuuride sätete taaskasutamine	165
Sõlme asetuste taaskasutus	166
Proksiobjektid	166
Skeletid ja mitmikesisindajad	166
3D-akna päis	168
Vaate menüü (View)	168
Valiku menüü (Select)	169
Objekti menüü (Object)	169
Režiimide nimekiri	169
Vaateakna varjutuste nimekiri (Viewport Shading)	169
Keskpunkti valik	170
Teisendamise (manipulaatorite) valikud	170
Kihtide valimine	170
Stseenile lukustamine	170
Võre külge nakkumine (Snap to Mesh)	170
Renderdamise nupud	170
Perspektiivsed ja paralleelsed vaated	171
Kirjeldus	171
Sätted	171
Tehnilised detailid	172
Perspektiivi definitsioon	172
Paralleelprojektsiooni definitsioon	172
Vaate pööramine	172
Kirjeldus	172
Seaded	172
Juhtkuul/plaadimängija	172
Vaate nihutamine	173
Kirjeldus	173
Vaate suurendamine	173
Kirjeldus	173
Suurendusala	173
Vaate joondamine	173
Joonda vaade	173
Kohalik ja globaalne vaade	174
Vaatekauguse ala	174
Kirjeldus	174
Näide	174
Kirjeldus	175
Vaate omaduste paneel	175

Kirjeldus	175
Vaade (View)	175
Ese (Item)	176
Esitus (Display)	176
Taustapilt	176
Seaded	176
Kiirvalikud	178
Kihtidega töötamine	180
Kihtide vaatamine	180
Stseenile lukustamine	180
Mitu kihti	180
Objektide kihtide vahel liigutamine	180
Kihtide animeerimine	181
Objektide kihtidele jagamise näide	181
Kirjeldus	182
Näited	182
Kirjeldus	185
Valikud	185
Kirjeldus	186
Sätted	186
Kirjeldus	187
Valikud	187
Kirjeldus	189
Teisendamissätete paneel	191
Sätted objekti režiimis	191
Teisendamissätete lukustamine	191
Teisenduste kinnitamine	192
Teisenduste tühistamine	192
Manipulaatorite kasutamine	193
Manipulaatori sätted	193
Teisendussuuna valimine	194
Valikud	195
Kohandatud suunad	196
Teisendussuundade paneel	196
Telgede lukustamise viisid	198
Telgede lukustamise režiimid	198
Kuidas telgesid lukustada	198
Kasutades klaviatuuri	198
Kasutades hiirt	199
Keskpunkti valimine	200
Objektide keskpunktid	200
Objektide keskpunkti liigutamine	200
Objekti režiimis	201
Muutmisrežiimis	201
Üksikvalik	201
Mitmikvalik	201
Objekti režiimis	203
Muutmisrežiimis	203
3D-kursori asetamine	205
Teisendamine	205
Objektirežiimis	206
Muutmisrežiimis	206
Teisendamine	206
Objekti režiimis	207
Muutmisrežiimis	207
Nakkumisrežiimid	209
Nakkumise element (Snap Element)	209
Nakkumise sihtmärk (Snap Target)	209
Pöörde joondamine	209
Objektirežiim	210
Muutmisrežiim	210
Valikud	210
Näited	211
Mis on rasvakriit?	213

Vabajoone tõmbamine	214
Kirjeldus	214
Vihjed	214
Sirgjoone tõmbamine	214
Kirjeldus	214
Kustukumm	214
Kirjeldus	214
Sätted	214
Vihjed	214
Visandiseansid	214
Kirjeldus	214
Vihje	215
Jagatud rasvakriidi sätted	215
Joonistamise sätted	215
Piirangud	215
Joonistamise tundlikus	215
Digilaua kasutajatele	215
Tasapindade joonistamine	215
Lisainformatsioon versiooni 2.5x kohta	215
Visandite animeerimine	216
Sibulakoore meetod	216
Visandite ajastuse muutmine	216
Visandite kopeerimine	216
Objektide tüübid	219
Objekti keskpunkt ehk kese ehk asukohapunkt (Object Centers)	219
Objekti keskme muutmine	219
Objektide kustutamine	220
Objektide ühendamine (Join Objects)	220
Valikud ja aktiivne objekt	221
Punktvalik	221
Nelinurga ehk piirdkastiga valimine	221
Kirjeldus	221
Näide	221
Vihje	222
Lassoga valimine	222
Kirjeldus	222
Kasutamine	222
Menüüga valimine	222
Vali grupeeritud (Grouped)	222
Kirjeldus	222
Valikud	222
Vali lingitud (Linked)	223
Kirjeldus	223
Valikud	223
Lisavõimalused	223
Teised menüü võimalused	224
Objektide allutamine	226
Allutamine tippudele	226
Valikud	226
Alluvobjekti liigutamine	226
Alluvussuhte muutmine ja tühistamine	226
Allutamise näide	227
Soovitused	227
Objektide eraldamine	227
Valikud	227
Objektide grupeerimine	227
Valikud	227
Grupeeritud objektide valimine	228
Valikud	228
Kirjeldus	230
Erinevad järgimise variandid	230
Teljepiiranguga järgimine (Track To Constraint)	230
Lukustatud järgimine (Locked Track Constraint)	230
Lihtne järgimine (Damped Track Constraint)	230

Vana järgimise piiraja (Old Track)	230
Vihjed	231
Mittesobiv järgimine või seadistused	231
Kirjeldus	232
Näited	232
Lingitud duplikaadid (Linked Duplicates)	232
Kirjeldus	232
Näited	232
Protseduuriline duplitseerimine (Procedural Duplication)	233
Lingitud andmekogud (Linked Library Duplication)	233
Soovitused	233
DupliTipud kui paigutamise tööriist	234
Alustamine	234
Ümberpaigutamine	234
Suunamine	234
DupliTipud modelleerimise tööriistana	235
Loe edasi	235
Asja põhimõte	236
Mõõtkava (Scale)	236
Üldine kasutamine	237
DupliGrupid ja dünaamiline linkimine	237
Näide	237
DupliGrupi objekti reaalseks muutmine	237
Näited	238
Välised viited	238
Muutmisrežiimi sisenemine	240
Visualiseerimine	240
Tööriistariul (Tool Shelf)	240
Omaduste ala (Properties Shelf)	241
Tipud (Vertices)	242
Servad (Edges)	242
Küljed (Faces)	242
Silmused (Loops)	242
Serva silmused (Edge Loops)	242
Küljesilmused (Face Loops)	243
Kirjeldus	244
Valikud	244
Lisad	245
Valimise režiimid (Selection Modes)	246
Valimise tööriistad vaate päises	246
Valimisrežiimi hüpikmenüü	246
Valimisrežiimide vahel lülitumine	246
Valitud elemendid ja valimisrežiimide vahetamine	246
Lihtne valimine	247
Valikusse lisamine	247
Piirkonna alusel valimine (Selecting Elements in a Region)	247
Piirdkastiga valimine (Border select)	248
Ringkujuline valikuala	248
Vaba käega märgitud ala (Lasso)	248
Täiendavad valimise tööriistad	249
Põhilised tööriistad	249
Erilisemad tööriistad	249
Valimine sarnasuse alusel (Select Similar)	249
Silmuste valimine (Selecting Loops)	250
Serva- ja tipusilmuse valimine (Edge Loop and Vertex Loop Selection)	250
Näide	250
Küljesilmuse valimine (Face Loop Selection)	251
Näited	251
Servarõnga valimine (Edge Ring Selection)	251
Näide	252
Silmus piirkonnaks ja vastupidi (Loop to Region and Region to Loop)	252
Näide: Silmus piirkonnaks (Loop to Region)	252
Näide: Piirkond silmuseks (Region to Loop)	253
Valikurežiimid (Select Modes)	254
Valikurežiimi hüpikmenüü	254
Valikurežiimi päisenupud	254

Validud elemendid pärast valikurežiimi muutmist	254
Servasilmused (Edge Loops)	256
Servarõngad (Edge Rings)	256
Tööriistade tüübid	257
Võre tööriistadele ligi pääsemine	257
Võre tööriistade palett	258
Menüüd	258
Teisendamise tööriistad	259
Liiguta, pööra, muuda suurst	259
Peegeldamine (Mirror)	259
Serva nihutamine (Edge Slide)	259
Suru kokku/lamenda (Shrink/Flatten)	259
Serva pööramine (Edge Flip)	259
Geomeetria moonutamine	259
Suru kokku/lamenda (Shrink/Flatten)	259
Lükka/tõmba (Push/Pull)	259
Vääna (Warp)	259
Näide	259
Aja kiiva (Shear)	260
Kerakujuliseks (To Sphere)	261
Näide	261
Müra (Noise)	261
Silu (Smooth)	261
Kustutamine	263
Liitmine	263
Kolmnurkade nelinurkadeks muutmise	263
Uute servade ja külgede loomine	264
Eenda (Extrude)	264
Tahkesta (Solidify)	264
Tükelda (Subdivide)	264
Silmuse lisamine (Loop Cut)	264
Nuga (Knife Tool)	264
Duplitseeri (Duplicate)	264
Neljatipuliste külgede kolmetipulisteks muutmise	265
Vurr (Spin)	265
Kruvi (Screw)	265
Eraldamine (Separate)	266
Poolitamine (Split)	266
Lõhestamine (Rip)	266
Võrega nakkumine (Mesh Snapping)	267
Sobitamise töötleja (Shrinkwrap Modifier)	267
Voolimisrežiim	268
Voolimise pintslid	268
Mitmikresolutsiooni töötlejaga voolimine (Multires Modifier)	269
Voolimise omaduste paneel	269
Pintsli menüü (Brush)	269
Tõmbe menüü (Stroke)	269
Kõvera menüü (Curve)	270
Tekstuuri menüü (Texture)	270
Sümmeetria menüü (Symmetry)	270
Valikute menüü (Options)	271
Välimuse menüü (Appearance)	271
Tööriista menüü (Tool)	271
Võre peitmine ja näitamine	271
Klaviatuuri kiirvalikud	271
Mille poolest on tipugruppide kasutamine hea?	273
Tipugruppide loomine ja kustutamine	273
Valimine ja valiku tühistamine	274
Tippude tipugruppi lisamine ja eemaldamine	274
Tipugruppide haldamine	274
Maalimise paneel	276
Kaalude tööriistad (Weight Tools)	277
Pintslitõmbed (Stroke)	277
Mõjukõvera kuju (Curve)	277
Valikud	278
Väljanägemine (Appearance)	278
Tööriist (Tool)	278

Külgede valiku maskimine (Face Selection Masking)	278
Kaalude maalimine skeleti elustamisel	279
Kaalude maalimine osakeste genereerimiseks	280
Sile varjutamine (Smooth shading)	281
Võre osade silumine	281
Automaatne silumine (Auto Smooth)	281
Servalõhestaja töötleja (Edge Split Modifier)	281
Võre geomeetria silumine	282
Võre muutmise tööriistad	282
Töötlejad (Modifiers)	282
Võrdlus polügonaalsete võrega	284
Kõvera tööriistade leidmine	284
Visualiseerimine	284
Elementide peitmine	284
Kõvera struktuur	285
Kontrollpunktid ja segmendid	285
Kõvera resolutsioon	285
Kahe- ja kolmemõõtmelised kõverad	285
Suletud ja avatud kõverad	285
Bezier' kõverad	285
NURBS-kõverad	286
Sõlmed (Knots)	286
Järk	287
Kaal	287
Eelseadistatud kaalud	288
Primitiivid (Primitives)	288
Lihtne kõverate muutmine (liigutamine, pööramine, suuruse muutmine)	289
Keerulisemad teisendamise tööriistad	289
Bezier' kontrollpunktide sätted	289
NURBSide kontrollpunktide sätted	289
Uute segmentide lisamine	289
Kõverate sulgemine ja avamine	289
Duplitseerimine (Duplication)	290
Elementide kustutamine	290
Kõverate ühendamine	290
Tükeldamine	290
Muuda suunda (Switch Direction)	291
Teised eriomadused	291
Tüübi muutmine	291
Mitmesugune muutmine	291
Topoloogia muutmine	291
Kõvera valimine	291
Valiku menüü	292
Iga n. (Every Nth)	292
Vali/Tühista valik esimene/vimane (Select/Deselect First/Last)	292
Vali järgmine/eelmine (Select Next/Previous)	292
Rohkem ja vähem (More/Less)	292
Kasutajaliides	293
Näidis	294
Kõvera eendamine	295
Eendamine (Extrusion)	295
Lihtne eendamine	296
Keerulisem eendamine	296
Näited	297
Pinna tööriistade leidmine	299
Kuvamine	299
Pinna struktuur	299
Kontrollpunktid, read ja ruudustik	299
Pinna resolutsioon	300
Suletud ja avatud pinnad	300
Sõlmed	300
Järk	300
Kaal	301
Eelseadistatud kaalud	301
Primitiivid	302

Valikumenüü (Select)	303
Iga n. (Every Nth)	303
Kontrollpunktide rida (Control Point Row)	303
Rohkem ja vähem (More/Less)	303
Pinna muutmine	303
Lihtne pindade muutmine (liigutamine, pööramine, suuruse teisendamine)	304
Keerulisemad teisendamise tööriistad	304
NURBSide kontrollpunktide sätted	304
Lisamine ja eendamine	304
Näited	304
Pindade avamine ja sulgemine	305
Duplitseerimine (Duplication)	305
Elementide kustutamine	305
Näide	306
Pindade ühendamine	306
Näited	306
Tükeldamine (Subdivision)	306
Vurr (Spin)	307
Muuda suunda (Switch Direction)	307
Teised eriomadused	307
Konverteerimine	307
Topoloogia muutmine (Retopology)	307
Mitmesugune muutmine	307
Märkused	308
Teksti valimine	308
Teksti muutmine	308
Teksti sisestamine	309
Šriftid (fondid)	309
Šriftifailide (fontide) laadimine ja muutmine	309
Objektid kui fondid	310
Fontide formaatimine	310
Rasvane, kald- ja allajoonitud kiri	310
Suurtähtede ja väiketähtede kasutus	311
Joondamine	311
Joonda tekst (Align)	311
Tähevahed (Spacing)	311
Nihe (Offset)	311
Erimärgid (Special Characters)	312
Teksti raamid (Text Boxes)	312
Tekstiraami suurus (Frame size)	312
Raami lisamine/kustutamine	312
Näidis: Teksti voolamine	312
Näide: mitme veeru kasutamine	313
Mitme materjali kasutamine	313
Näited	313
Tekst kõveral	313
Näide	313
Kõvera kuvamise valikud	314
Resolutsioon (Resolution)	314
Täide (Fill)	314
Tekstuurid	314
Kantimine ja eendamine	314
Muutmine (Modification)	314
Kantimine (Bevel)	315
Konverdi tekst tekstiobjektiks	315
Primitiivid	316
Kuvamine	316
Metapalli sätted	316
Resolutsioon	317
Lävi (mõju)	317
Värskendamine	317
Meta struktuur	318
Tehnilised detailid	318
Alusstruktuur	318
Meta kuju	320
Jäikus (Stiffness)	320

Negatiivne mõju (Negative Influence)	320
Elementide peitmine	321
Elementide kustutamine	321
Konverteerimine (Conversion)	321
Objektide perekonnad	321
Näited	322
Vihjed	322
Kuvamise seaded	323
Moodustajad	324
Moontajad	324
Simulatsioonid	325
Kasutajaliides	326
Pinu	326
Valikud	328
Kasutamine	328
Kirjeldus	329
Üldised sätted	329
Tipu kaalude muutmise töötleja (Vertex Weight Edit Modifier)	329
Sätted	330
Tipu kaalude segamise töötleja (Vertex Weight Mix Modifier)	330
Sätted	331
Tipu kaalude läheduse töötleja (Vertex Weight Proximity Modifier)	331
Sätted	331
Näited	332
Sihtobjekti kauguse kasutamine	332
Sihtobjekti geomeetria kauguse kasutamine	332
Tekstuuri ja laotuskövera kasutamine	333
Vaata ka	334
Kirjeldus	335
Seaded	335
Vihjeid	336
Teisenduse arutamine	336
Näited	337
Mehaanilised	337
Fraktaalsed	337
Orgaanilised	338
Õppetükid	338
Õppetükid	338
Kirjeldus	339
Mis on kantimine (bevel)?	339
Valikud	339
Width - Laius	339
Ainult tipud	340
Limit Method - Piiramisviis	340
Õppetükid mujal	341
Kirjeldus	342
Valikud	342
Vaata ka	342
Kirjeldus	343
Valikud	343
Kirjeldus	344
Valikud	344
Näited	344
Lihtne tasand	344
Lihtsustatud silinder	344
Kirjeldus	346
Valikud	346
Näited	346
Kirjeldus	347
Valikud	347
Kirjeldus	348
Valikud	348
Vihjed	349
Peegeldaja (Mirror modifier) kasutamine koos pinnatükeldajaga (Subdivision Surface Modifier)	349
Joondamine peegeldamiseks	349
Valikud	350
Kirjeldus	351
Valikud	351
Kirjeldus	352

Valikud	352
Kirjeldus	353
Valikud	353
Pinnatükeldus (Subdivision Surfaces)	353
Kirjeldus	353
Valikud	353
Viikide teravused pinna jaotamisel (Weighted creases)	354
Kirjeldus	354
Valikud	354
Kirjeldus	355
Valikud	355
Kirjeldus	356
Valikud	356
Kirjeldus	357
Valikud	357
Kirjeldus	358
Valikud	358
Vaata ka	359
Kirjeldus	360
Valikud	360
Valikud	361
Vihjed	361
Näited/Õppetüki	361
Osakesed ja sõrestikud	361
Kirjeldus	363
Valikud	363
Vaata ka	363
Kirjeldus	364
Valikud	364
Kirjeldus	366
Valikud	366
Kirjeldus	367
Valikud	367
Valikud	368
Kirjeldus	369
Valikud	369
Tehnilised detailid ja soovitus	370
Kirjeldus	371
Töö järjekord	371
Riidesimulatsioonide loomine	371
Riide paneel	371
Materjal	372
Sumbumine (Damping)	372
Kinnitamine (Pinning)	372
Kokkupõrked	372
Kokkupõrke seaded	372
Kokkupõrked iseenesega	373
Jagatud kihid	373
Võreobjektide kokkupõrked	373
Riide/objekti põrked	373
Võreobjekti töötlejate pinu	374
Riide vahemälu	374
Põrke eeltöötlus	374
Vahemälu oleva simulatsiooni muutmine	374
Probleemide lahendamine	374
Näited	374
Simulatsiooni kasutamine võre vormimiseks/voolimiseks	375
Riide silumine	375
Riie skeletil	375
Riide kasutamine pehmete kehadena	375
Riie ja tuul	375
Valikud	376
Pehme keha ja riide vastastikune mõju	377
Vastastikmõju jõuväljadega	377
Näited	377
Vihjed	377

Kirjeldus	378
Valikud	378
Kirjeldus	379
Protsess	379
Valikud	379
Toimimispiirkond (Domain)	380
Lahutus (Resolution)	380
Eelvaate lahutus (Preview Res)	380
Alguse ja lõpu aeg	380
Kuvamise kvaliteet (Disp.-Qual)	381
Eeltöötuse kataloog (Bake Directory)	381
Nupp "Bake" (eeltöötle)	381
Märkused...	381
Teised toimimispiirkonna valikud	381
Vedelik	383
Takistus	383
Sissevool	384
Väljavool	384
Osake	384
Juhtimine	385
Vedeliku omaduste muutuste animeerimine	385
Tehnilised detailid	386
Vihjed	387
Piirangud ja lahendused	387
Vaata ka	388
Tänuõnad	388
Simulatsiooni tööpõhimõte	389
Ocean Modifier - ookeanitöötaja	389
Kirjeldus	389
Geomeetria valikud	389
Simulaatori valikud	389
Lainete valikud	389
Simulatsiooniandmete loomise valikud	390
Eeltöötus (baking)	390
Andmefailid	390
Vahu eeltöötus	390
Eeltöötuse valikud	391
Ajalugu	391
Kirjeldus	392
Ülevaade	392
Valikud	392
Loomine	393
Esitus	393
Trajektoorie kasutamine	393
Vaata ka	396
Tööjärjekord	397
Osakeste süsteemi loomine	397
Osakeste süsteemide tüübid	398
Ühised valikud	398
Lingid	398
Töö käik	399
Toimimispiirkond (Domain)	399
Loo toimimispiirkond	399
Valikud	399
Suitsugrupid	400
Suits kõrge lahutusega	400
Suitsuvälja kaalud	400
Sissevool (Inflow)	400
Loo voolu objekt	400
Seaded	401
Kokkupõrked	401
Jõud	401
Suitsu renderdamine	401
Loo materjal	401
Lisa tekstuur	402
Suitsusimulaatori laiendamine: Tuli!	404
Suitsu simulatsioonide eeltöötus	405
Tõrkeotsing	406
Välised lingid	406
Pehmete kehade kasutamise tüüpiline stsenaarium	407

Pehme keha loomine	407
Simulatsiooni kvaliteet	408
Vahemälu ja eeltöötlus	408
Suhtlus reaajas	408
Vihjed	409
Lingid	409
Tajumise piirangud	410
Globaalsed mõjud	410
Valgustuse sätted	410
Töö valgusega	410
Materjalide tühistamine valgustuse algseadistamiseks	410
Objekti andmed (Object Data)	413
Eelvaade (Preview)	413
Valgusti (Lamp)	413
Kirjeldus	414
Valguse nõrgenemiskõverad	414
Lineaame/ruutsõltuvus (Lin/Quad Weighted)	414
Lineaame (Linear)	414
Ruutsõltuvus (Quadratic)	414
Lineaarse ja ruutsõltuvusliku segamine	415
Mõlema liuguri nullimine	415
Kokkuvõtte pildis	415
Kasutaja määratud nõrgenemiskõver	416
Pöördvõrdeline ruutsõltuvus (Inverse Square)	417
Pöördvõrdeline lineaarne (Inverse Linear)	417
Ühtlane (Constant)	418
Kera	418
Näidised	418
Kaugus	418
Pöördvõrdeline ruutsõltuvus	419
Kera	419
Näpunäited	420
Valikud	421
Valgustigrupid	423
Materjalid	423
Stseen	423
Vaata ka	423
Valgustid: kiirtejälituse meetodil (Ray-traced) heidetud varjud	424
Valgustid: puhvervarjud	425
Vaata ka	426
Mis on kvaasi-Monte Carlo?	428
Valgusti parameetrid	430
Üldised parameetrid	430
Varjud	430
Varjud kiirtejälituse meetodil	430
Valgusti parameetrid	431
Üldised parameetrid	431
Varjude parameetrid	431
Valgusvihu parameetrid	431
Puhverdatud vari	432
Kohtvalgusti varjud kiirtejälituse meetodil	432
Kohtvalgusti ruumilised efektid (volumetric effects)	433
Varjupuhvri tüübid	434
"Klassikaline" ja "keskmistatud klassikaline" varjude genereerimine	434
Korrapärase varjugeneraator	436
Sügav varjugeneraator	436
Üldised parameetrid	436
Näpunäited	437
Valikud	439
Ühised valikud	439
Varjud	439
Pinna kuju	439
Pindvalguse kiirtejälitusega varjud	440
Varju testkiired (Shadow samples)	440
Testkiirte generaatorite tüübid	440
Tehnilised detailid	440
Näpunäited	441
Seaded	442

Tehnilised detailid	442
Valikud	443
Ühised parameetrid	443
Taeva & atmosfääri paneel	443
Varjude paneel	443
Päikese varjud kiirtejälitusega	443
Näpunäited	444
Valikud	446
Taevas (Sky)	446
Atmosfäär (Atmosphere)	447
Näited	447
Taevas (Sky)	448
Atmosfäär (Atmosphere)	450
Näpunäited ja piirangud	451
Näide	453
Vaata ka	453
Valikud	454
Kiirtejälitus	454
Hindamine (Approximate)	456
Ühised parameetrid	456
Tehnilised detailid	457
Näpunäited	457
Varjutajad (Shaders)	459
Hajusvarjutus (Diffuse Shading)	459
Läikevarjutus (Specular Shading)	459
Pegelduvus (Reflectivity)	460
Läbipaistvus (Transparency)	460
Läbikumavus (Translucency)	460
Pinna sisehajumine (Subsurface Scattering)	460
Hõõgumine (Incandescence)	460
Materjali jagamine mõne teise objektiga	461
Materjali linkimine objekti või objekti andmetega	462
Materjali valikud	462
Kirjeldus	463
Valikud	463
Näited	463
Kontekstimenüü	464
Materjalide pesad	464
Mitmikmaterjalid	464
Materjalide nimetamine ja linkimine	464
Materjali tüüp	465
Materjali omaduste ülevaade	465
Pind- ja traatmaterjalid	465
Eelvaade (Preview)	465
Hajusus (Diffuse)	465
Läige (Specular)	466
Varjutamine (Shading)	466
Läbipaistvus (Transparency)	466
Pegeldus (Mirror)	466
Pinna sisehajuvus (Subsurface Scattering)	466
Karv (Strand)	466
Valikud (Options)	466
Vari (Shadow)	467
Mahtmaterjalid	467
Halomaterjalid	467
Tipugruppide haldamine	468
Tippude määramine tipugruppi	468
Tippudele materjalide määramine	468
Seaded	469
Tehnilised detailid	469
Märkus	469
Lambert	469
Seaded	469
Oren-Nayar	470
Seaded	470
Multifilmi (Toon)	470
Seaded	471

Minnaert	471
Seaded	471
Fresnel	471
Seaded	472
Kirjeldus	473
Seaded	473
Tehnilised andmed	473
CookTorr	473
Kirjeldus	474
Valikud	474
Phong	474
Kirjeldus	474
Valikud	474
Planeedi atmosfäär	474
Blinn	474
Kirjeldus	475
Valikud	475
Multifilmi (Toon)	475
Kirjeldus	475
Valikud	475
Vihje	475
WardIso	475
Kirjeldus	476
Valikud	476
Kirjeldus	478
Sätted	478
Värviribad	479
Valikud	479
Kirjeldus	481
Seaded	481
Näidised	482
Fresnel	482
Kiirtejälitusega läbipaistvus (Raytraced Transparency)	484
Sätted	484
Näited	485
Murdumise indeks	485
Fresnel	485
Sügavus	486
Vihjed	486
Läbipaistvad varjud	486
Igapäevaste materjalide IOR-väärtused	487
Pinna sisehajumise sisse lülitamine	489
Seaded	489
Kuidas luua omaenda sisehajuvusega materjali	490
Näited	490
Nahk	490
Loe lisaks	490
Karvade varjutamine	491
Karvadele tekstuuri lisamine	492
Karvade renderdamise lihtsustamine	492
Mahtudele tekstuuri andmine	494
Sätted	494
Tihedus	494
Varjutamine	494
Hajumine	494
Asümmeetria	495
Kandumine	495
Kiirgus (Emission)	495
Peegeldus	496
Vihjed	496
Valgustus (Lighting)	496
Läbipaistvus	497
Integratsioon (Integration)	497
Sätted (Options)	497
Näited	497
Sätted	498
Sõlmeredaktorisse sisenemine	499

Materjali nuppude alt sõlmematerjalide sisse lülitamine	499
Välised lingid	500
Sisenemine sõlmeredaktorisse	501
Sõlmede aktiveerimine	501
Sõlmeredaktori akna tegevused	501
Sõlmeredaktori päis	502
Vaate, valimise ja lisamise päisemenüüd	502
Tööriistad	502
Tüübi valimise nupp	503
Sõlmede kasutamise päisenupp (Use Nodes)	503
Sõlmede lisamine ja paigutamine	503
Pesad	503
Pesade ühendamine ja lahtiühendamine	504
Sõlme juhtelemendid	504
Sõlme suuruse muutmine	505
Sõlmekõverad	505
RGB-kõverad	506
Kõverapunktide valimine	506
Kõverate muutmine	506
Vaate muutmine	506
Eritööriistad	506
Sõlmegrupid	506
Grupeerimine	506
Sõlmegruppide muutmine	507
Sõlmegrupi lõhkumine	507
Sõlmegruppide lisamine	507
Kokku voltimise nupud	508
Sõlme suuruse muutmine	509
Pesad	509
Kõverad	510
RGB-kõverad	510
Kõverapunktide valimine	510
Kõverate muutmine	511
Vaate muutmine	511
Eritööriistad	511
Sõlmede lisamine	512
Sõlmede paigutamine	512
Sõlmede ühendamine	512
Sõlmede lahtiühendamine	512
Sõlmede duplitseerimine	512
Grupeerimine	513
Sõlmegruppide muutmine	513
Sõlmegrupi lõhkumine	513
Sõlmegruppide lisamine	513
Materjalisõlm (Material)	515
Väljund	515
Sisend	515
Juhikud	515
Materjalisõlme kasutamine läikega	516
Väärtuse sõlm (Value)	516
RGB sõlm (RGB)	516
Tekstuurisõlm (Texture)	516
Geomeetria sõlm (Geometry)	517
UV-piili kasutava geomeetria sõlme näide	517
Lisaomadustega materjal (Extended Material)	517
Sisend	517
Väljund	518
Kaamera andmed (Camera Data)	518
Mix - sega	520
RGB kõverad (RGB Curves)	520
Pöördväärtusta (Invert)	521
Toon-küllastus-heledus (Hue Saturation Value)	521
Normaalisõlm (Normal)	522
Normaalsõlme näidis	522
Normaali kasutamine mügarlike tekstuuride peegeldamiseks	522
Normaalisõlme kasutamine spetsiaalsete valgusallikate loomiseks	523
Laotamissõlm (Mapping)	523
Laotamissõlme näidis	524
Vektorite kõverad (Vector Curves)	524

Värvülemineku sõlm (ColorRamp)	525
Värvieemaldussõlm (RGB to BW)	525
Matemaatika (Math)	525
Vektormatemaatika (Vector Math)	525
Väärtuse kokkusurumine (Squeeze Value)	525
Eralda RGB kanalid ('Separate RGB)	525
Kombineeri RGB kanalid (Combine RGB)	525
Sätted	527
Pintsel (Brush)	527
Tööriist (Tool)	527
Pintslitõmme (Stroke)	527
Kõverad (Curves)	528
Välimus (Appearance)	528
Valikud (Options)	528
Seaded	529
Halo paneel	529
Efektid	529
Peegelduse paneel (Flare)	530
Objektiivi peegeldus	530
Halo tekstuur	531
Näited	531
Maatriksekraan	531
Materjalitekstuurid	533
Maaailma tekstuurid (World)	534
Pintslite tekstuurid (Brush)	534
Tekstuuri konteksti valimine	535
Tekstuuri andmetüübi valimine	535
Tekstuuri pesad	535
Uude tekstuuuri pesasse uue tekstuuuri andmebloki loomine	535
Täis pesasse uue tekstuuuri andmebloki loomine	535
Tekstuuri andmebloki jagamine täis pesaga	536
Tekstuuride pinu	537
Tekstuuri andmeblokk	537
Tekstuuri tüüp	537
Eelvaade (Preview)	537
Värvid (Colors)	538
Laotus (Mapping)	538
Mõju (Influence)	538
Ühised omadused	540
Müra baas	540
Nabla	541
Soovitused	541
Valikud	542
Valikud	543
Valikud	544
Valikud	545
Tehnilised detailid	545
Valikud	546
Valikud	547
Tehnilised detailid	547
Valikud	548
Tehnilised detailid	548
Valikud	549
Tehnilised detailid	549
Valikud	550
Valikud	551
Tehnilised detailid	551
Piltidega tekstuuridest	552
Töö käik	552
Piltide ja materjalide kasutamine	552
Pilditekstuuri loomine	553
Näited	553
UV-tekstuuride üksteise peale panemine	553
Materjalide segamine ja ühendamise	554
Läbipaistvuse (Alpha Transparency) kasutamine	554
UV-tekstuurid vs protseduurilised tekstuurid	555
Sätted	555
Pilt (Image)	555

Pildi lugemine (Image sampling)	556
Pildi laotamine (Image Mapping)	558
Tekstuurisõlmede kasutamine	560
Mitme väljundi kasutamine	560
Vaata ka	560
Koordinaadid	561
Projektsioon	561
Koordinaatide pärimine ülemobjektidelt	562
Koordinaatide nihe, skaala ja teisendus	562
UV-kaardi selgitus	563
Kartograafiline näide	563
Poolkera näide	563
UV-redaktor	564
Päis	564
Omaduste paneel	564
UV-ruumis navigeerimine	564
Esituse valikud	565
UV-laotuse kasutamise eelised	565
UV-kaartidest	566
Alustamine	566
Töö käik	566
Laotuse tüübid	566
Põhilaotused	567
Kuup	567
Silinder ja kera	567
Projektsioon vaatest	568
UV-algväärtustamine	568
Täiuslikumad lahtilõiked	568
Lahtilõige õmbluste abil	568
Õmbluste märkimine	569
Lahtilõige	570
Tark UV-projektsioon	570
Valguskaart	571
Järgi aktiivseid nelinurki	571
Mitme külje lahtilõige	572
UV-laotuse ülekandmine	573
Mitme UV-laotuse kasutamine	573
UV-tekstuuride nimekiri	573
UV-elementide valimine	574
Valikurežiimid	574
UV-elementide ümberpaigutamine	575
Telje lukustamine	575
Teisenduste keskpunktid	575
Mõjualaga muutmine	575
Nakkumine	575
Keevitus ja joondamine	575
Peegeldus	575
Nõelumine	575
Venituse minimeerimine	575
Külgede peegeldamine ja UV-pööre	575
Kinnitamine	576
UV-laotuse optimeerimine	576
UV-kaartide kombineerimine	576
Keskmise saare mõõtkava	577
Saarte pakkimine	577
Piiramine pildiäärtega	577
Iteratsioonid ja täiustamine	577
Laotuse täiustamine	577
Tekstuuride taaskasutamine	578
UV-laotusjoonise eksportimine	579
UV-tekstuuride kasutamine	579
Piltide laadimine ja salvestamine	580
Uued pildid	580

Testvõrgu kasutamine	580
Pildi seadistused	581
Pilditekstuuri muutmine	582
Pildi asendamine eelrenderduse tulemusena	582
Pildi loomine/muutmine välises programmis	582
Pildi muutmine tekstuurimaalijaga	582
Piltide salvestamine	582
Tekstuuri määramine külgedele	582
Aktiivse pildi asendamine	582
Piltide pakkimine Blenderi faili	582
Pind ja traatmaterjalid	585
Hajusus (Diffuse)	585
Läige (Specular)	585
Varjutamine (Shading)	585
Geomeetria (Geometry)	585
Segamine (Blending)	585
Mahulised materjalid	586
Halo materjalid	586
Kirjeldus	588
Töö käik	588
Kirjeldus	589
Valikud	589
Näpunäited	589
Kirjeldus	591
Seaded	591
Tekstuurid	591
Kirjeldus	593
Seaded	593
Läbipaistvus	593
Näide	593
Kirjeldus	594
Sätted	594
Sinu esimene skelett	596
Skeletiobjekt	596
Skeletipeatüki ülevaade	596
Skeleti paneel Skeleton (kõik režiimid)	598
Esituspaneel Display (kõik režiimid)	598
Luugruppide paneel Bone Groups (poseerimisrežiim)	598
Pooside teegi paneel Pose Library (poseerimisrežiim)	598
Kummituse paneel Ghost (kõik režiimid)	599
iTASC parameetrite paneel iTASC parameters (kõik režiimid)	599
Liikumisradade paneel Motion Paths (poseerimisrežiim)	599
Täiendavate omaduste paneel Custom Properties (kõik režiimid)	599
Luu paneelide ülevaade	599
Suhete paneel Relations (muutmisrežiim)	600
Esituspaneel Display (objektirežiim)	600
Moonutuste paneel Deform (kõik režiimid)	600
Täiendavate omaduste paneel Custom Properties (kõik režiimid)	600
Teisenduste paneel Transform (muutmis- ja poseerimisrežiim)	600
Teisenduste lukustamise paneel Transform Locks (poseerimisrežiim)	601
Pöördkinemaatika paneel Inverse Kinematics (poseerimisrežiim)	601
Luude visualiseerimine	602
Luude omadused	603
Luude jäikus	603
Luude mõju	603
Esituse paneel (Display)	605
Luu tüübid	605
Oktaeedriline luu (Octahedron)	605
Pulgakujuline luu (Stick)	606
B-luu esitus (B-Bone)	606
Ümbriku esitus (Envelope)	606
Täiendavad esitused	606
Määratud kujuga luud	606
Seaded	607
Skeletikihid	607
Luukihtide näitamine ja peitmine	608
Kaitstud kihid	608
Luukihid	608
Luude liigutamine kihtide vahel	608

Luude peitmine	608
Luude ahelad (Chains)	610
Luude hierarhiiliste suhete määramine	610
Luu otspunktide valimine	611
Valiku vastupidiseks pööramine	611
Seotud luude otste valimine	611
Luude valimine	611
Seotud luude valikust eemaldamine	612
Luude lisamine	614
Lisamismenüü (Add)	614
Eendamine (Extrusion)	614
Hiirenupud	614
Liigeste vahe täitmine (Fill between joints)	616
Duplitseerimine (Duplication)	617
Luude kustutamine	617
Tavaline kustutamine	617
Ühendamine (Merge)	618
Luude tükeldamine (Subdividing)	618
Luude lukustamine	619
X-pegeldusega (X-Axis Mirror) muutmine	619
Luude eraldamine uude skeleti	619
Luude teisendamine	620
Raadius ja möötkava ümbriku esitusrežiimis	620
BMöötkava (ScaleB) ja ümbrik	620
Luu suund	621
Luu pööramine	621
Omadused	622
Luu jäikuse seaded	622
Ahela muutmine	623
Luude nimetamine	624
Nimetamisreeglid	624
Luu nime ümber pööramine	625
Automaatne luude nimetamine	625
Ahelate joonistamine	626
Sirged segmendid	626
Vabakäe segmendid	627
Tõmmete valimine	627
Kustutamine	627
Tõmmete muutmine	627
Žestid	627
Lõikamine	628
Kustuta	628
Pööra tagurpidi	628
Luudeks konverteerimine	628
Fikseeritud (Fixed)	629
Kohalduv (Adaptative)	629
Pikkus (Length)	629
Mall (Template)	629
Sihtpunkti uuesti märkimine (Retarget)	629
Poseerimise osa ülevaade	631
Vaata ka	631
Automaatne IK (Automatic IK)	632
IK piirajad (IK Constraints)	632
IK splain (Spline IK)	632
Kasutamise põhipunktid	632
Parameetrid ja seaded	632
Pööre (Roll)	632
Nihe (Offset)	632
Jämedus (Thickness)	632
Nõuanded hea tulemuse saamiseks	632
Kasutamise põhipunktid	634
Parameetrid ja seaded	634
Pööre (Roll)	634
Nihe (Offset)	634
Jämedus (Thickness)	634
Nõuanded hea tulemuse saamiseks	634
Kirjeldus	635
Piirajate kasutamine	635

Luude piiramine	635
Olemasolevad piirajad	635
Teisendamise piirajad	635
Järgimise piirajad	636
Suhtepiirajad	636
Piirajate päis	637
Piirajate sätted	637
Sihtmärk	637
Piiraja koordinaatsüsteem (Space)	638
Mõju	638
Piiraja lisamine ja eemaldamine	639
Kirjeldus	640
Valikud	640
Valikud	641
Kirjeldus	642
Valikud	642
Kirjeldus	643
Valikud	643
Kirjeldus	644
Valikud	644
Kirjeldus	645
Valikud	645
Kirjeldus	646
Valikud	646
Kirjeldus	647
Valikud	647
Kirjeldus	648
Valik	648
Loe lisaks	648
Kirjeldus	649
Valikud	649
Kirjeldus	651
Valikud	651
Kirjeldus	652
Valikud	652
Kirjeldus	653
Valikud	653
Kirjeldus	654
Valikud	654
Kirjeldus	655
Valikud	655
Loe lisaks	655
Valikud	656
Kirjeldus	657
Valikud	657
Kirjeldus	658
Valikud	658
Märkused	658
Kirjeldus	660
Valikud	660
Näpunäited	660
Näide	660
Kirjeldus	662
Valikud	662
Kirjeldus	663
Valikud	663
Kirjeldus	665
Valikud	665
Näide	665
Loe lisaks	665
Kirjeldus	666
Valikud	666
Kirjeldus	668
Valikud	668
Peatükid	669
Ajatelje elemendid	670
Praegune kaader	670
Võtmekaadrid (Keyframes)	670

Markerid (Markers)	670
Vaate muutmise	670
Vaate menüü (View)	670
Muutmise	671
Kaadri menüü (Frame)	671
Taasesitus	671
Taasesituse menüü (Playback)	671
Päise nupud	672
Poosimarkerid (Pose markers)	674
Visualiseerimine	674
Tavapärane	674
Monteerija (Sequencer)	674
3D-vaade	674
Markerite loomine ja muutmise	674
Markerite loomine	674
Markerite valimine	675
Markerite nimetamine	675
Markerite liigutamine	675
Markerite duplitseerimine	675
Markerite kustutamine	675
Võtmekaadrite tekitamine	676
3D-vaates	676
Omaduste paneelil	676
Animatsiooniredaktorites	676
Võtmekaadrite muutmise	677
Interpoleerimise põhimõte	679
Interpolatsiooni tüübid	679
Aja suund	679
Kõvera valimine	679
Valimise tööriistad	680
Kõverate muutmise	680
Teisendused	680
Kopeerimine	680
Kusutamine	680
Nakkumine	680
Peegelda (Mirror)	681
Puhastamine ja silumine	681
Interpolatsioon ja ekstrapolatsioon	681
Interpolatsioon (Interpolation)	681
Ekstrapolatsioon (Extrapolation)	682
Sanga tüübid	682
Laotus (Mapping)	683
Juhtijate loomine	683
Juhtijate muutmise	683
Juhtija omadused	683
Juhtijate tüübid	683
Muutujad (Variables)	683
Üksikomadus (Single Property)	684
Teisenduskanal (Transform Channel)	684
Näide	684
Pöörde erinevus (Rotational Difference)	685
Kaugus (Distance)	685
Avaldisjuhtijad (Expression Drivers)	685
Valiidsed avaldised	685
Sisseehitatud võimalused ja aliased	686
Blenderi tekst nimega pydrivers.py	686
Näidis	686
Lingid	687
Trajektoorid	688
Ribad	688
Tegevusribade loomine	688
Üleminekuribade loomine	688
Metaribade loomine	688
Ribade muutmise	688
Ribade korduvkasutamine	689

Riba omadused	689
Ribade ümbemimetamine	689
Aktiivne rada	689
Aktiivne riba	689
Aktiivne tegevus	690
Väärtuse arvutamine	690
Riba töötlejad	690
Töötleja lisamine	691
Töötlejate tüübid	691
Generaator	691
Sisseehitatud funktsioonid	691
Ümbrik	691
Tsüklid	691
Müra	692
Python	692
Piirid	692
Astmeline	692
Animatsiooni juhtimine piirajatega	693
Näide	693
Piirajate mõju animeerimine	693
Hierarhiline meetod	694
Trajektoori piiraja	694
Klammerduspiiraja	694
Gravitatsioon	695
Jõuvälja loomine	696
Ühised väljade seaded	696
Kahanemine (Falloff)	696
Väljade tüübid	697
Jõud (Force)	697
Tuul (Wind)	697
Keerisväli (Vortex Field)	697
Magnetväli (Magnetic)	697
Harmooniline (Harmonic)	697
Laeng (Charge)	697
Lennard-Jones	698
Tekstuur väli (Texture Field)	698
Näited	698
Suunav kõver (Curve Guide)	698
Parv (Boid)	699
Turbulents (Turbulence)	699
Takistus (Drag)	700
Lingid	700
Valikud	701
Pehme keha ja riide vastastikune mõju	702
Vastastikmõju jõuväljadega	702
Näited	702
Vihjed	702
Tööjärjekord	703
Osakeste süsteemi loomine	703
Osakeste süsteemide tüübid	704
Ühised valikud	704
Lingid	704
Valikud	705
Emissiooni asukoht	705
Jaotuse seaded	705
Kiirus (Velocity)	707
Emiteeriv geomeetria (Emitter Geometry)	707
Emiteeriv objekt (Emitter Object)	707
Pöörlemine	707
Ühised füüsika seaded	708
Füüsika puudub	708
Välja kaalud (Field Weights)	708
Jõuväljad (Force Fields)	708
Jõud	710
Kokkupõrge	710
Integreerimine	710

Ülesseadmine	711
Võtmed	711
Ajastamine	711
Füüsika	712
Võitlus	713
Liit (Alliance)	713
Suunajad ja mõjutajad (Deflectors and Effectors)	713
Parve aju (Boid Brain)	713
Reeglite arvestamine	714
Seaded	715
Vedeliku omadused	715
Keerulisemad	715
Vedrud	715
Esitus 3D-vaates (Viewport Display)	717
Värv (Color)	717
Renderdamise seaded (Render Settings)	717
Puudub (None)	717
Halo	718
Joon (Line)	718
Trajektoor (Path)	718
Objekt (Object)	719
Grupp (Group)	719
Tahvlid (Billboard)	719
Vahemälu kasutamine	722
Eeltöötlus	722
Kasvamine	723
Kujundamine	723
Animeerimine	723
Renderdamine	723
Valikud	723
Emissioon	723
Karvade dünaamika	723
Esitus	723
Alamosakesed	724
Renderdamine	724
Kasutamine	724
Seaded	725
Material	725
Sumbumine (Damping)	725
Kvaliteet	725
Seaded	726
Efektid	726
Karedus (Roughness)	727
Vääne (Kink)	727
Näited	728
Osakeste režiimi kasutusviisid	729
Osakeste režiimi kasutamine	729
Punktide valimine	729
Võtmepunktide või osakeste liigutamine	729
Osakeste peegeldamine	729
Peitmine/näitamine	730
Valikurežiimid	730
Osakesterežiimi omadused	730
Pintsell (Brush)	730
Valikud	730
Pehme keha kasutamise tüüpiline stsenaarium	732
Pehme keha loomine	732
Simulatsiooni kvaliteet	733
Vahemälu ja eeltöötlus	733
Suhtlus reaalsajas	733
Vihjed	734
Lingid	734
Näide	735
Seaded	735
Pehme keha paneel	735
Jõuväljad	735

Aerodünaamika	735
Eesmärgi kasutamine	735
Servade seaded	737
Kokkukukkumise vältimine	737
Põrked teiste objektidega	739
Näited	739
Põrgete arvutamine	739
Korralik põrge	740
Kokkupõrge iseenesega	740
Põrkav kuup	742
Tulemus	742
Kirjeldus	743
Töö järjekord	743
Riidesimulatsioonide loomine	743
Riide paneel	743
Materjal	744
Sumbumine (Damping)	744
Kinnitamine (Pinning)	744
Kokkupõrked	744
Kokkupõrke seaded	744
Kokkupõrked iseenesega	745
Jagatud kihid	745
Võreobjektide kokkupõrked	745
Riide/objekti põrked	745
Võreobjekti töötlejate pinu	746
Riide vahemälu	746
Põrke eeltöötlus	746
Vahemälu oleva simulatsiooni muutmine	746
Probleemide lahendamine	746
Näited	746
Simulatsiooni kasutamine võre vormimiseks/voolimiseks	747
Riide silumine	747
Riie skeletil	747
Riide kasutamine pehmete kehadena	747
Riie ja tuul	747
Kirjeldus	748
Protsess	748
Valikud	748
Toimimispiirkond (Domain)	749
Lahutus (Resolution)	749
Eelvaate lahutus (Preview Res)	749
Alguse ja lõpu aeg	749
Kuvamise kvaliteet (Disp.-Qual)	750
Eeltöötuse kataloog (Bake Directory)	750
Nupp "Bake" (eeltöötle)	750
Märkused...	750
Teised toimimispiirkonna valikud	750
Vedelik	752
Takistus	752
Sissevool	753
Väljavool	753
Osake	753
Juhtimine	754
Vedeliku omaduste muutuste animeerimine	754
Tehnilised detailid	755
Vihjed	756
Piirangud ja lahendused	756
Vaata ka	757
Tänuõnad	757
Töö käik	758
Toimimispiirkond (Domain)	758
Loo toimimispiirkond	758
Valikud	758
Suitsugrupid	759
Suits kõrge lahutusega	759
Suitsuvälja kaalud	759
Sissevool (Inflow)	759
Loo voolu objekt	759
Seaded	760

Kokkupõrked	760
Jõud	760
Suitsu renderdamine	760
Loo materjal	760
Lisa tekstuur	761
Suitsusimulaatori laiendamine: Tuli!	763
Suitsu simulatsioonide eeltöötlus	764
Tõrkeotsing	765
Välised lingid	765
Töötleja aktiveerimine	766
Tüübid	766
Vaata lisaks	766
Peamine paneel	767
Allika paneel (Source)	767
Võre maht (Mesh Volume)	767
Lähedus (Proximity)	767
Võre maht + lähedus (Mesh Volume + Proximity)	768
Objekti kese (Object Center)	768
Osakeste süsteem (Particle System)	768
Kiiruse paneel (Velocity)	768
Lainete paneel (Waves)	769
Peamine paneel	770
Lisasätete paneel (Advanced)	770
Ühised sätted	770
Värv (Paint)	770
Nihe (Displace)	771
Lained (Waves)	771
Kaal (Weight)	772
Väljundi paneel (Output)	772
Efektide paneel (Effects)	772
Puhvri paneel (Cache)	772
OpenMP (Mac OSX)	775
Vähene eelrenderduse jõudlus	775
Lahendus	775
Applikripti kasutamise sammud	775
Ülevaade	776
Renderdusfarm	776
Renderdustööde ühendamine	776
Renderduse seadistuste paneel	777
Render	777
Kihid	777
Mõõtmed	777
Sakisilumine (Anti-Aliasing)	777
Liikumishägu (Motion Blur)	777
Varjutamine (Shading)	777
Väljund	778
Jõudlus	778
Järeltöötlus (Post Processing)	778
Tempel (Stamp)	778
Eelrenderdamine (Bake)	778
Uue kaamera lisamine	779
Aktiivse kaamera vahetamine	779
Kaamera seaded	779
Lääts (Lens)	779
Esitus (Display)	780
Kompositsioonijooned	780
Kaameraga navigeerimine	781
Kaamera vaatega kohakuti liigutamine	781
Kaamera paigutamine vaate abil	781
Roll, Pan, Dolly ja Track	781
Kaameraga lendamisrežiimis sihtimine	781
Kolmepunktiline renderdus	783
Kahepunktiline renderdus	783
Kahe koondumispunktiga renderduse tegemiseks:	784
Ühe koondumispunktiga renderdamine	784
Ühe koondumispunktiga renderdamine	784

Paralleelprojektsioonis renderdamine (Orthographic Rendering)	785
Isomeetriline renderdamine	785
Vaateväli ja läätse fookuskaugus	786
Suurendamine Blenderis	786
Sügavusteravus arvutigraafikas	786
Renderdatud pildi kuvamine	787
Salvestamine	787
Esitusviisid	787
Värvide haldamine (Color Management)	788
Sakisilumine (Antialiasing)	788
Säriaeg (Exposure)	788
Sügavusteravus (Depth of Field)	788
Liikumishägu (Motion Blur)	788
Valikud	789
Filtreerimine	789
Filtri suurus	790
Näited	790
Töövoog	792
Kaadrite kaupa renderdamine	792
Valikud	793
Vihjed	793
Valikud	794
Eelrenderdamise viis	794
Full Bake (täielik renderdus)	794
Ambient Occlusion (hajusvalgustus)	794
Shadow (vari)	794
Normals (normaalid)	795
Textures (tekstuudid)	795
Displacement (nihutus)	795
Emission (valguskiirgus)	795
Alpha (alfa)	795
Mirror Color and Intensity (peegli värv ja intensiivsus)	795
Specular Color and Intensity (läike värv ja intensiivsus)	795
Lisavalikud	796
Töövoog	796
Süntaks	797
Renderdamise valikud:	797
Animatsiooni valikud:	797
Akna valikud:	797
Mängumootorispetsiifilised valikud:	797
Muud valikud:	798
Näited	798
Pildiks renderdamine	798
Videoks renderdamine	798
Mõõtmed (Dimensions)	799
Formaadid	799
Väljundi paneel Output	799
Failide asukohad	799
Faili tüüp	800
Pakkimine	800
Kanalid	800
Pildifaili vormingud	800
Videomonteerijaga renderdamine	801
Renderdamine pildijadaks	801
Renderdamine videoks	801
Piksli külgede suhe	802
Värviküllastus (Saturation)	802
Poolkaadritena renderdamine	802
Valikud	802
Videoformaadid	803
Pakkimine edasijõudnutele	804
Video seadistused (Video Settings)	804
Heli seadistused (Audio Settings)	805
Nipid	806
Renderduskihid komposiitori kasutamisel	808
Kas kihid või käigud (Layers või Passes)?	808

Renderduskihtide kasutamine	808
Aktiveerimine ja nimetamine	808
Uue renderduskihi tekitamine	808
Stseeni kihtide seaded	808
Ümberülitused (Overrides)	808
Sisalduvad valikud (Include Options)	809
Renderduskäigud (Passes)	809
Näited	809
Ainult mõningate objektide renderdamine	809
Ainult valitud objektidele äärejoone tegemine	809
Renderduskäigud detailsemalt	810
Renderduskäikude valimine	810
Renderduskäikude vahelejätmine	811
Renderduskäikude kasutamine	811
Varjude toonimine	811
Kaudvarju (Ambient Occlusion) komposiitimine	812
Vektorhäguga varjud	812
Kokkuvõte	813
Valikud	814
Examples	814
Virttoonimine (Dithering)	815
Inimsilm	817
Filmis	817
Arvutigraafikas	817
Liikumishägu Blenderis	817
Sämplitud	817
Vektorhägu	817
Näited	817
Vihjeid	818
Eesmärgid	819
Allalaadimine	819
Juhised	819
Kasutajaliides	819
Isand (Master)	819
Orjad (Slaves)	819
Klient	819
Käsurida	819
Lisainfo	819
Seadistused	819
Peaserver	820
Ori	820
Klient	820
Märkused ja teadaolevad vead	821
Koormuse tasakaalustamine	821
Erandid	821
Esmased prioriteedid (kriteeriumid)	821
Mis on veel plaanis teha	822
Tehnilised täpsustused	822
Võimaluste loetelu	822
API funktsioonide soovinimekiri	822
Valikud	824
Üldvalgustus	824
Näited	825
Konsooli väljund	825
Renderdamine ilma väljundita	825
Renderdatud pilt	826
Märkused	826
Valguse hulk	826
Taevakuppel (SkyDome)	827
Alustamine	828
Õppetükid	828
Lisainfo	828
Perspektiiv (Perspective)	832
Paralleelprojektsioon(Orthographic)	832
Sügavusteravus (Depth of Field)	832
Vaate piiramine (Clipping)	832
Pinnavarjutaja (Surface Shader)	833
Mahu varjutaja (Volume Shader)	833
Nihkelaotus (Displacement)	833

Energia säilitamine	833
Tüüp (Type)	834
Tükeldamine (Subdivision)	834
Terminoloogia	835
BSDFi parameetrid	835
3D-vaateakna joonistusrežiimid	837
Tekstuuri omadused	837
Maalimine ja UVde muutmine	837
Pinnavarjutaja (Surface Shader)	838
Mahu varjutaja (Volume Shader)	838
Nipid	838
Punktvalgus (Point Lamp)	839
Pindvalgus (Area Lamp)	839
Päikesevalgus (Sun Lamp)	839
Varjutajad (Shaders)	840
Tekstuudid (Textures)	840
Lisasõlmed	840
Kahesuunalist hajutamise funktsioon (BSDF)	841
Hajuspeegeldus (Diffuse)	841
Läbikumav (Translucent)	841
Läikepeegeldus (Glossy)	841
Läbipaistev (Transparent)	842
Klaas (Glass)	842
Samet (Velvet)	843
Kiirgus (Emission)	843
Taust (Background)	844
Kohahoidja (Holdout)	844
Segamine (Mix) ja liitmine (Add)	844
Pildi tekstuur (Image Texture)	846
Keskkonna tekstuur (Environment Texture)	846
Taeva tekstuur (Sky Texture)	846
Müratekstuur (Noise Texture)	846
Lainetekstuur (Wave Texture)	846
Voronoi tekstuur (Voronoi Texture)	847
Musgrave'i tekstuur (Musgrave Texture)	847
Värvülemineku tekstuur (Gradient Texture)	847
Maagia tekstuur (Magic Texture)	847
Ruudustiku tekstuur (Checker Texture)	847
Väärtus (Value)	849
RGB	849
Geomeetria (Geometry)	849
Tekstuuri koordinaadid (Texture Coordinates)	849
Atribuut (Attribute)	849
Laotus (Mapping)=	849
Kihtide kaalud (Layer Weight)	850
Fresnel	850
Valguse teekond (Light Path)	850
Kiirte tüübid	851
Pörkekontroll	851
Läbipaistvus (Transparency)	851
Kiirte nähtavus	851
Stseeni sätted	852
Sämplid (Samples)	852
Põrked (Bounces)	852
Läbipaistvus (Transparency)	852
Kaustika (Caustics)	852
Materjali sätted	852
Maailma sätted	852
Sissejuhatus	854
Seadistamine	854
CUDA	854
Vanemad kaardid	854
OpenCL	854
Korduma kippuvad küsimused	854
=Miks Blender renderdamise ajal ei vasta?	854
Miks stseeni, mida tavaline protsessor renderdab, graafikaardiga ei renderdata?	854
Kas saan renderdamiseks kasutada mitut graafikaarti?	854
Kas mitu graafikaarti suurendaksid kasutatava mälu hulka?	854
Mis renderdab kiiremini – NVidia või AMD, CUDA või OpenCL?	855
Veateated	855
Unsupported GNU version! gcc 4.5 and up are not supported!	855

CUDA Error: Invalid kernel image	855
CUDA Error: Out of memory	855
The NVIDIA OpenGL driver lost connection with the display driver	855
Sõlmede mõisted	856
Sõlmed (Nodes)	856
Nuudlid (Noodles)	856
Sõlmede grupid (Node Groups)	856
Sõlmeredaktorisse sisenemine ja sõlmede aktiveerimine	856
Näited	857
Sisenemine sõlmeredaktorisse	858
Sõlmede aktiveerimine	858
Sõlmeredaktori akna tegevused	858
Sõlmeredaktori päis	859
Vaate, valimise ja lisamise päisemenüüd	859
Tööriistad	859
Tüübi valimise nupp	860
Sõlmede kasutamise päisenupp (Use Nodes)	860
Sõlmede lisamine ja paigutamine	860
Pesad	860
Pesade ühendamine ja lahtiühendamine	861
Sõlme juhtelemendid	861
Sõlme suuruse muutmine	862
Sõlmekõverad	862
RGB-kõverad	863
Kõverapunktide valimine	863
Kõverate muutmine	863
Vaate muutmine	863
Eritööriistad	863
Sõlmegrupid	863
Grupeerimine	863
Sõlmegruppide muutmine	864
Sõlmegrupi lõhkumine	864
Sõlmegruppide lisamine	864
Kokku voltimise nupud	865
Sõlme suuruse muutmine	866
Pesad	866
Kõverad	867
RGB-kõverad	867
Kõverapunktide valimine	867
Kõverate muutmine	868
Vaate muutmine	868
Eritööriistad	868
Sõlmede lisamine	869
Sõlmede paigutamine	869
Sõlmede ühendamine	869
Sõlmede lahtiühendamine	869
Sõlmede duplitseerimine	869
Grupeerimine	870
Sõlmegruppide muutmine	870
Sõlmegrupi lõhkumine	870
Sõlmegruppide lisamine	870
Renderduskihtide sõlm (Render Layers)	872
Alpha-väljundi kasutamine	872
Valikulised väljundid	873
Z-kanali kasutamine	873
Kiirusvektori ühenduspesa (Speed) kasutamine	874
Pildi lugemise sõlm (Image)	874
Pildi kanalid	874
Renderduskäikude salvestamine ja laadimine	875
Pildi suurus	875
Animatsioonid	875
Genereeritud pildid	876
Märkused	876
Videote lõikamine ja üleminekud	876
Lõike ülemineku loomine sõlmede abil	876
Sisse sulatamine	877
Sulatamine ühest klipist teise	877
Värvide karmimine	877
Tekstuursõlm (Texture)	878
Näide	878

Väärtuse genereerimise sõlm (Value)	879
RGB sõlm (RGB)	879
Näide	879
Ajasõlm (Time)	879
Näiteid ajasõlmedest	880
Näited ja soovitus	880
Kuvamissõlm (Viewer)	881
UV/pildiredaktori kasutamine kuvamissõlmega	881
Komposiitkuva (Composite)	881
Komposiitpildi salvestamine	881
Võrdluskuva sõlm (SplitViewer)	881
Salvestussõlm (File Output)	882
Statistikasõlm (Levels Node)	882
RGB-kõverate sõlm (RGB Curve)	883
Valikud	883
Näited	883
Värvikorrektuur kõverate abil	883
Värvikorrektuur musta/valge tasemetega	884
Efekid	884
Segamissõlm (Mix)	884
Näited	885
Kontrasti võimendamine segamifunktsiooniga	886
Segamissõlme kasutamine pildi vesimärgistamiseks	887
Vesimärgi lisamine pildile	887
Vesimärgistatud pildi dekodeerimine	887
Dodge ja Bum (ala- ja ülesäritamine) segamifunktsioonide kasutamine	887
Toonimissõlm (Hue Saturation)	888
Toonimissõlme nipid	888
Toonimise näide	888
Heleduse ja kontrasti sõlm (Bright/Contrast)	888
Märkused	889
Gamma	889
Negatiiv (Invert)	889
Valikud	890
AlphaOver sõlm	890
Näited	890
Veidrad halod ja ümbrisjooned	890
Sügavuse kombineerimise sõlm (Z-Combine)	891
Näited	891
Värvitasakaalu sõlm (Color Balance)	892
Toonikorrektuuri sõlm (Hue Correct)	892
Tooniteisenduse sõlm (Tone Map)	892
Normaalisõlm (Normal)	894
Vektori ülekandekõverad (Vector Curves)	894
Ülekandefunktsiooni sõlm (Map Value)	894
Map Value sõlme kasutamine korrutamiseks	894
Normaliseerimissõlm (Normalize)	895
Filtrisõlm (Filter)	896
Udustamissõlm (Blur)	896
Valikud	896
Suunatud udustamise sõlm (Directional Blur)	897
Valikud	897
Näide	897
Bilateraalse udu sõlm (Bilateral Blur)	897
Sisendid	897
Valikud	897
Näited	898
Vektorhägu sõlm (Vector Blur)	898
Näited	899
Teadaolevad vead	899
Laiendamis-süvistamissõlm (Dilate/Erode)	899
Näide	899
Sügavusudu sõlm (Defocus)	899
Kaamera seaded (Camera Settings)	899
Sõlme sisendid	900
Sõlme seaded	900
Näited	901
Näpunäited	901
Värviteisenduse sõlm (ColorRamp)	903
ColorRamp sõlme kasutamine maski loomiseks	903
ColorRamp sõlme kasutamine pildi värvimiseks	903
RGB mustvalgeks muutmise sõlm (RGB to BW)	903
Läbipaistvuse määramise sõlm (Set Alpha)	904
SetAlpha sõlme kasutamine pildi mustaks sulatamisel	904
SetAlpha sõlme kasutamine tiitrite pildile sulatamiseks	904

SetAlpha sõlme kasutamine pildi toonimiseks	904
ID Mask sõlm	905
Näide	905
Matemaatika sõlm (Math)	905
Näited	906
Z-maski loomine käsitsi	906
Siinusfunktsiooni kasutamine pulseeriva pildi loomiseks	906
Pildikanali skaleerimine	906
Kvantiseerimine ja värvipaleti piiramine	907
Teooria	907
Tegelikkus	908
Kokkuvõtteks	908
Kombineerimissõlmed (Combine/Separate)	908
Separate/Combine RGBA sõlm	909
Näited	909
Separate/Combine HSVA sõlm	909
Separate/Combine YUVA sõlm	910
Separate/Combine YCbCrA sõlm	910
Alpha Convert sõlm	910
Erinevusmask (Difference Key)	911
Lihtne näide	911
Mitme maskisõlme kooskasutus	912
Chroma Key sõlm	912
Värvimaski sõlm (Color Key)	913
Heledusmaski sõlm (Luminance Key)	913
Näide	913
Värvilekke eemaldaja (Color Spill)	914
Kanalimask (Channel Key)	914
Kaugusmask (Distance Key)	914
Nihutussõlm (Translate)	916
Näide: kuidas kasutada nihutussõlme (Translate) kerivate tiitrite loomiseks	916
Näide: animeeritud taust	916
Näide: kaamera rappumine	917
Pööramissõlm (Rotate)	918
Mõõtkava teisendamise sõlm (Scale)	918
Peegeldamissõlm (Flip)	918
Pikslinihutussõlm (Displace)	918
Näide	919
Map UV sõlm	919
Näited	919
Lõikamissõlm (Crop)	920
Läätsemoonutuse sõlm (Lens Distortion)	920
Monteerija kasutamine	922
Ribade lisamine	922
Videote või piltide lisamine	923
Stseeni lisamine	923
Heli lisamine	924
Efektide lisamine	924
Plugin-efektide lisamine	924
Riba omadused	925
Riba muutmise paneel (Edit Strip)	925
Efektiriba (Effect Strip)	925
Riba sisend (Strip Input)	926
Filter	926
Asendusriba omaduste paneel (Proxy Strip Properties)	926
Heli (Sound)	926
Stseen (Scene)	926
Vaate muutmine	926
Kerimine	927
Vaaterežiimid	927
Stseeni eelvaade	927
Vaate seaded	927
Värskenda vaadet	927
Ribade valimine	928
Ribade liigutamine ja muutmine	928
Kaadriga nakkumine	928
Eralda pildid ribadeks	928
Ribade muutmine	928
Kopeeri ja aseta	929
Metaribad	929
Pildi eelvaade	930
Heleduse lainevorm	930
Värvilisuse vektroskoop	931
Tulpdiagramm	932

Liida (Add)	933
Lahutamine (Subtract)	933
Üleminek (Cross) ja gammapõhine üleminek (Gamma Cross)	933
Mustaks sulandumine (Fade to Black)	933
Korrutamine (Multiply)	934
Alpha Over, Alpha Under ja Alpha Over Drop	934
Liikuv üleminek (Wipe)	935
Kuma (Glow)	935
Teisendamine (Transform)	936
Värv (Color)	936
Kiiruse määraja (Speed Control)	936
Aegluubi (Slow Motion) efekti saavutamine	936
Kadrite sobitamine	937
Video kaadrisageduse muutmine	937
Mitmikkaamera valija (Multicam Selector)	938
Kohanduskiht (Adjustment Layer)	938
Seaded	939
Helimontaaž videomontaažihaldajas	939
Audioradadega töötamine	939
Audioraja omaduste animeerimine	939
Väljund	939
Sissejuhatus	941
Üldine informatsioon	941
Alustamine - Wiki õppetükid	941
Alustamine - välised lingid	941
Lisade installeerimine	943
Failide asukohad Blenderi versioonis 2.5	943
Sisse ja välja lülitamine	943
Arendusjuhendid	943
Mis on programmeerimine?	944
Mis on Python?	944
Pythoni interpretaator	944
Hello World	944
Välised lingid	945
Veebilehed	945
Videoõppetükid	945
Sisenemine sisseehitatud Pythoni konsooli	946
Esmane ülevaade konsooli keskkonnast	946
Automaatne lõpetamine töötamas	946
Enne moodulite kallal nokitsemist...	947
Näited	947
bpy.context	947
Proovi seda!	947
bpy.data	948
Proovi seda!	948
Harjutus	948
bpy.ops	948
Proovi seda!	948
Tekstiakna teised kasutusvõimalused	950
Näide	950
Ülesanne	950
Geomeetria	952
Kuidas ma saan API abil võreobjekti luua?	952
Kuidas ma API abil töötleja lisan?	952
Kuidas ma leian Bezier' kõvera kontrolltipu maailma koordinaadid?	952
Kuidas ma valin kõvera kontrollpunktid või tühistan selle valiku?	952
Meetod 1	952
Meetod 2	953
Materjalid	953
Kuidas linkida võret/objekti materjaliga?	953
Ise täiendamine	953
Kuidas ma endatehtud kiirvalikud automatiseerin?	953
Skriptid	954
Mängumootori kasutamine	955
1 Omadused (Properties)	956

2 Seotud objekt(id) (Associated Objects)	956
3 Lingid (Links)	957
4 Sensorid (Sensors)	957
5 Kontrollerid (Controllers)	957
6 Käiturid (Actuators)	957
Omaduste paneel	958
Omaduste tüübid	958
Omaduste kasutamine	958
Omaduse sensor (Property Sensor)	958
Omaduse käitur (Property Actuator)	959
Sensori vaate valikud	960
Üldised seaded	960
Üldised sensori seaded	960
Sensori päis	960
Sensori impulsiriba	961
Käituri sensor (Actuator sensor)	963
Pidev sensor (Always sensor)	963
Põrke sensor (Collision sensor)	963
Viivituse sensor (Delay sensor)	963
Juhtkangi sensor (Joystick sensor)	963
Klaviatuuri sensor (Keyboard sensor)	964
Teate sensor (Message sensor)	964
Hiire sensor (Mouse sensor)	964
Läheduse sensor (Near sensor)	965
Omaduse sensor (Property sensor)	965
Radari sensor (Radar sensor)	966
Juhuslikkuse sensor (Random sensor)	966
Kiire sensori (Ray sensor)	966
Puute sensor (Touch sensor)	966
Kontrollerite võrdlev tabel	968
Kontroller AND (ja)	968
Näide	968
Kontroller OR (või)	968
Näide	968
Kontroller XOR (välistav või)	968
Näide	968
Kontroller NAND (mitte ja)	969
Näide	969
Kontroller NOR (mitte või)	969
Näide	969
Kontroller XNOR (välistav mitte või)	969
Näide	969
Avaldise kontroller (Expression)	969
Pythoni kontroller	969
Skriptirežiim (Script Mode)	969
Moodulrežiim (Module Mode)	969
Muutujad	970
Toimingud	970
Matemaatilised tehted	970
Loogikatehted	970
Tingimuslaused (if)	970
Näited	970
Dünaamika (Dynamics)	973
Jälita (Track To)	973
Asenda võre (Replace Mesh)	973
Lõpeta objekt (End Object)	973
Lisa objekt (Add Object)	973
Liikumishägusti	974
Sisseehitatud 2D-filtrid	974
Kasutaja loodud filtrid	975
Tüüp (Type)	976
Märkused kasutamise kohta	976
Märkus kasutamise kohta	977
Näide	978
Toiming (Operation)	980

Märkus kasutamise kohta	980
Märkus kasutamise kohta	981
Vaikekaamera	982
Kaamera allutamine objektile	982
Tipule allutamine	982
Objekt kui kaamera	982
Kaamera läätse nihutamine	982
Loe lisaks	982
Stereo seaded	983
Stereorežiimid	983
Nelikpuhver (Quad Buffer)	983
Üleval-all (Above-Below)	983
Ülerealaotus (Interlaced)	983
Anaglüüf (Anaglyph)	983
Kõrvuti (Side by Side)	983
Vertikaalne ülerealaotus (Vinterlaced)	983
Kuppelkaamera seaded	984
Kalasilma režiim (Fisheye)	984
Eest lõigatud kuppel (Front-Truncated Dome)	985
Tagant lõigatud kuppel (Rear-Truncated Dome)	985
Kuuplaotuse režiim (Cube Map)	985
Kerapanoraam (Spherical Panoramic)	986
Väändevõre andmed (Warp Data Mesh)	986
Näidisfailid	987
Seaded	988
Sätted	989
Põrkepiirid (Collision Bounds)	989
Objekti füüsika, kui aktiivset maailma ei ole	989
Tegija (Actor)	990
Kummitus (Ghost)	990
Nähtamatu (Invisible)	990
Raadius (Radius)	990
Anisotroopne (Anisotropic)	990
Seaded	992
Omadused	992
Kiirus (Velocity)	992
Summutus (Damping)	992
Lukusta liikumine ja pööre (Lock Translation and Rotation)	992
Põrke piirid (Collision Bounds)	992
Seaded	994
Omadused	994
Parvpõrked (Cluster Collision)	994
Põrke piirid (Collision Bounds)	994
Kuidas see töötab	995
Kuidas neid kasutada	995
Mängusisene juhtimine	995
Piirangud	996
Sätted	997
Põrke piirid (Collision Bounds)	997
Kuidas see töötab	999
Mängu ettevalmistus	999
Esimene näide	999
Materjali leidmine	999
Tekstuuri loomine	999
Muuda tekstuur püsivaks	1000
Alustekstuuri loomine	1000
Allika ettevalmistamine	1000
Video mängimine	1001
Video staatuse kontrollimine	1001
Keerukamad võimalused	1001
Lae endale demo	1002
Keerukamad demod	1002
Väline dokumentatsioon	1004
Kogukond	1004
Arendus	1004
Mängude arendamise tööriistad ja vahendid	1004
Python	1004

Sissejuhatus

Mis on Blender?

- [Sissejuhatus](#)
- [Blenderi ajalugu](#)
- [Blenderi litsents](#)
- [Blenderi kogukond](#)
- [Käsiraamatu info](#)

Blenderi paigaldamine

- [Sissejuhatus](#)
- [Python](#)
- [Paigaldamine Windowsis](#)
- [Paigaldamine Linuxile](#)
- [Paigaldamine Macile](#)
- [Paigaldamine teistesse operatsioonisüsteemidesse](#)
- [Blenderi seadistamine](#)
- [Turvalisus](#)

Alustamine

Kasutajaliides

- [Sissejuhatus](#)
- [Klaviatuur ja hiir](#)
- [Akende süsteem](#)
- [Paanide paigutamine](#)
- [Päised](#)
- [Tekstikonsooli aken](#)
- [Akende tüübid](#)
- [Ekraanid \(töökeskonna paigutused\)](#)
- [Stseenid](#)
- [Kontekstid](#)
- [Menüüd](#)
- [Paneelid](#)
- [Nupud ja juhtelemendid](#)

Peamised funktsioonid

- [Kiirrenderdus](#)
- [Ekraanipildid ja ekraanilindistus](#)
- [Vaikestseen](#)
- [Tühistamine ja uuesti tegemine](#)
- [Abi!](#)

Eelistuste seadistamine

- [Eelistuste seadistamine](#)
- [Kasutajaliides](#)
- [Muutmine](#)
- [Sisendid](#)
- [Lisamoodulid](#)
- [Teemad](#)
- [Failieelistused](#)
- [Süsteem](#)

Sinu esimene animatsioon

- [1/2: Staatiline piparkoogimehike](#)
- [2/2: Animeeritud piparkoogimehike](#)

Andmesüsteem ja failid

Tegevused failidega

- [Blenderi failide avamine](#)
- [Blenderi failide salvestamine](#)
- [COLLADA](#)

Blenderi andmesüsteem

- [Blenderi teek ja andmesüsteem](#)
- [Stseenid](#)
- [Stseenidega töötamine](#)
- [Ülevaateaken](#)
- [Lisamine ja linkimine](#)

Interaktsioonid 3D-ruumis

Sissejuhatus

- [Sissejuhatus navigeerimisse](#)

3D-ruumis navigeerimine

- [Sissejuhatus](#)
- [3D-vaate kasutamine](#)
- [3D-vaate valikud](#)
- [Kaameravaade](#)
- [Kihid](#)
- [Lokaalne ja globaalne vaade](#)

Teisendused

- [Sissejuhatus](#)
- [Algteadmised](#)
- [- Haara/liiguta](#)
- [- Pööra](#)
- [- Muuda mõõtkava](#)
- [Keerulisemad teisendused](#)
- [- Peegelda](#)

Teisenduste piiramine

- [Sissejuhatus](#)
- [Teisendamise sätted](#)
- [Manipulaatorid](#)
- [Teisendussuunad](#)
- [Telgede lukustamine](#)
- [Keskpunkt](#)
- [- Aktiivne objekt](#)
- [- Individuaalsed keskpunktid](#)
- [- 3D-kursor](#)
- [- Mediaanpunkt](#)
- [- Piirdkasti kese](#)
- [Nakkumine](#)
- [Võrega nakkumine](#)
- [Proportsionaalne muutmine](#)

Skitseerimine 3D-ruumis

- [Rasvakriidi tutvustus](#)
- [Visandamine](#)
- [Kihid ja animatsioon](#)
- [Visandi muutmine ruumiliseks](#)

Modelleerimine

Sissejuhatus

- [Sissejuhatus](#)

Objektid

- [Objektid](#)
- [Objektide valimine](#)
- [Objektide muutmine](#)
- [Objektide grupeerimine ja allutamine](#)
- [Järgimine](#)
- [Duplitseerimine](#)
- [- Dupli Tipud](#)
- [- Dupli Küljed](#)
- [- Dupli Grupp](#)
- [- Dupli Kaadrid](#)

Võre objektid

- [Võred](#)
- [- Võre struktuurid](#)
- [- Võre baasobjektid](#)
- [Valimine](#)
- [- Valitavad elemendid](#)
- [- Servade valimine](#)
- [Muutmine](#)
- [Lihne muutmine](#)
- [- Elementide kustutamine](#)

- [Duplitseerimise tööriistad](#)
- [Tükeldamise tööriistad](#)
- [Topoloogia muutmise tööriist](#)
- [Voolimisrežiim](#)
- [Tipugrupid](#)
- [Kaalude maalimine](#)
- [Võre silumine](#)

Kõverad

- [Kõverad](#)
- [Muutmine](#)
- [Keerukam muutmine](#)

Pinnad

- [Pinnad](#)
- [Pindade muutmine](#)

Tekstiobjektid

- [Tekstid](#)

Metaobjektid

- [Metad](#)
- [Muutmine](#)

Tühiobjektid

- [Tühiobjektid](#)

Grupiobjektid

Skriptid

Töötledjad ja moonutamine

Sissejuhatus

- [Töötledjate sissejuhatus](#)
- [Töötledjate pinu](#)

Muuda

- [UV-projektseerija](#)
- [Tipugrupi kaalude töötledja](#)

Moodusta

- [Reatöötledja](#)
- [Kantimistöötledja](#)
- [Kahendtöötledja](#)
- [Ehitaja](#)
- [Lihtsustaja](#)
- [Servalõhestaja](#)
- [Masktöötledja](#)
- [Peegeldaja](#)
- [Parralleelvõre](#)
- [Kruvitöötledja](#)
- [Paksuse lisaja](#)
- [Pinnatükeldaja](#)

Moonuta

- [Skeletiöötledja](#)
- [Lähendustöötledja](#)
- [Kõveratöötledja](#)
- [Nihutustöötledja](#)
- [Haaktöötledja](#)
- [Sõrestiktöötledja](#)
- [Võremoonutaja](#)
- [Sobitaja](#)
- [Moonutaja](#)
- [Silumistöötledja](#)
- [Väändtöötledja](#)
- [Lainetöötledja](#)

Simuleeri

- [Riie](#)
- [Kokkupõrge](#)
- [Plahvatus](#)
- [Vedelik](#)
- [Ookean](#)
- [Osake](#)
- [Osakeste süsteem](#)
- [Suits](#)
- [Pehme keha](#)

Valgustamine

Sissejuhatus

- [Sissejuhatus valgustamisse](#)

Valgused

- [Sissejuhatus](#)
- [Valguse omadused](#)
- [Valguse nõrgenemine](#)
- [Valguse tekstuurid](#)
- [Mida valgus mõjutab](#)
- [Valgustid teistes kontekstides](#)

Varjud

- [Sissejuhatus](#)
- [Varju omadused](#)
- [Kiirtejälitusega varju omadused](#)

Valgustid

- [Sissejuhatus](#)
- [Punktvalgustid](#)
- [Kohtvalgusti](#)
- [Puhverdatud varjud](#)
- [Halod?](#)
- [Pindvalgus](#)
- [Poolkera tüüpi valgustid](#)
- [Päikesevalgus](#)
- [Kiirtejälitusega varjud?](#)
- [Taevas ja atmosfäär](#)

Valgusvihud

- [Sissejuhatus](#)

Keskkonna valgustus ja varjutamine

- [Kaudvari](#)
- [Keskkonnavalgus](#)

Materjalid

Sissejuhatus

- [Materjalide sissejuhatus](#)

Kasutamine

- [Materjali määramine](#)
- [Materjali eelvaade](#)
- [Materjali sätted](#)
- [Mitmikmaterjalid](#)

Omadused

- [Hajusvarjutajad](#)
- [Läike varjutajad](#)
- [Üldvalguse efektid](#)
- [Värviüleminekud](#)
- [Kiirtejälitusega peegeldused](#)
- [Kiirtejälitusega läbipaistvus](#)
- [Pinna sisehajumine](#)
- [Karvad](#)
- [Mahulised materjalid](#)

Sõlmedest materjalid

- [Materjalisõlmed](#)
- [Sõlmeredaktor](#)
- [Sõlmede juhtelemendid](#)
- [Sõlmede kasutamine](#)
- [Sõlmegrupid](#)
- [Materjalisõlmede tüübid](#)
- [- Sisendi sõlmed](#)
- [- Väljundi sõlmed](#)
- [- Värvisõlmed](#)
- [- Vektorsõlmed](#)
- [- Konvertersõlmed](#)
- [- Dünaamilised sõlmed](#)

Tippude maalimine

- [Tippude maalimise kasutamine](#)

Halomaterjalid

- [Halomaterjalid](#)

Tekstuurid

Sissejuhatus

- [Sissejuhatus](#)
- [Tekstuuri määramine](#)
- [Tekstuuri sätted](#)

Tekstuuri tüübid

- [Tekstuuri tüübid](#)
- [Protseduurilised tekstuurid](#)
- [- Krohv](#)
- [- Maagia](#)
- [- Marmor](#)
- [- Moonutatud müra](#)
- [- Musgrave](#)
- [- Müra](#)
- [- Pilved](#)
- [- Puit](#)
- [- Sulandamine](#)
- [- Voronoi](#)
- [Pilditekstuurid](#)
- [Videotekstuurid](#)
- [Tekstuuri sõlmed](#)

Laotamine

- [Laotamise sätted](#)
- [UV-laotuse selgitus](#)
- [- Võre lahti lõikamine](#)
- [- UV-laotuste haldamine](#)
- [- UV-laotuse muutmine](#)
- [- Piltide kasutamine](#)

Mõju

- [Mõju](#)
- [- Materjal](#)
- [-- Faktuuri- ja normaalikaardid](#)
- [-- Nihkekaardid](#)

Maailm ja keskkonna efektid

Maailm

- [Sissejuhatus](#)
- [Maailma tagaplaan](#)

Keskkonna efektid

- [Udu](#)
- [Tähed](#)

Taageldamine

Sissejuhatus

- [Sissejuhatus taageldamisse](#)

Skeletid

- [Skeleti objektid](#)
- [Skeleti ja luude paneelid](#)
- [Luud](#)
- [Visualiseerimine](#)
- [Struktuur](#)
- [Valimine](#)
- [Muutmine](#)
- [Luud](#)
- [Omadused](#)
- [Skitseerimine](#)

Nahaga katmine

- [Sissejuhatus](#)

Poseerimine

- [Sissejuhatus](#)
- [Pöördkinemaatika](#)
- [- Splaini IK](#)

Piirajad

Sissejuhatus

- [Piirajate sissejuhatus](#)
- [Üldine kasutajaliides](#)
- [Piirajate pinu](#)

Teisendamise piirajad

- [Asukoha kopeerija](#)
- [Pöörde kopeerija](#)
- [Mõõtkava kopeerija](#)
- [Teisenduse kopeerija](#)
- [Kauguse piiraja](#)
- [Asukoha piiraja](#)
- [Pöörde piiraja](#)
- [Mõõtkava piiraja](#)
- [Mahuhaldaja](#)
- [Teisendus](#)

Järgimise piirajad

- [Klammerduspiiraja](#)
- [Sumbtrajektoori piiraja](#)
- [Pöördkinemaatika piiraja](#)
- [Lukustatud trajektoori piiraja](#)
- [Splaini IK piiraja](#)
- [Venitamispiiraja](#)
- [Järgimise piiraja](#)

Suhte piirajad

- [Tegevuse piiraja](#)
- [Alluvuse piiraja](#)
- [Põranda piiraja](#)
- [Raja järgimise piiraja](#)
- [Pöörlemiskeskme piiraja](#)
- [Jäiga keha liigendi piiraja](#)
- [Sobitamispiiraja](#)

Animeerimine

Sissejuhatus

- [Sissejuhatus animeerimisse](#)
- [Ajatelg](#)
- [Markerid](#)

Võtmekaadrid

- [Sissejuhatus](#)

Animatsiooniredaktorid

- [Animatsiooniredaktorid](#)
- [F-kõverad](#)
- [Juhtijad](#)
- [Mittelineaarse animatsiooni redaktor](#)
- [F-kõvera töötlejad](#)

Animatsioonitehnikad

- [Piirajate kasutamine](#)
- [Objektide liigutamine piki trajektoori](#)

Füüsikaline simulatsioon

Sissejuhatus

- [Sissejuhatus füüsikasse](#)

Dünaamika

- [Jõuväljad](#)
- [Kokkupõrked](#)

Osakesed

- [Osakesed](#)
- [Osakeste emissioon](#)
- [Füüsika](#)
- [- Newtoni füüsika](#)
- [- Võtmeosakesed](#)
- [- Parved](#)
- [- Vedelik](#)
- [Visualiseerimine](#)
- [Vahemälu & eeltöötlus](#)
- [Karvad](#)
- [- Karvade dünaamika](#)
- [Alamobjektid](#)
- [Tipugrupid](#)
- [Osakeste režiim](#)

Pehme keha

- [Sissejuhatus](#)
- [Välised jõud](#)
- [Sisised jõud](#)
- [Kokkupõrked](#)
- [Lihtsad näited](#)

Riie

- [Sissejuhatus](#)

Vedelikud

- [Vedelik](#)
- [Suits](#)

Dünaamiline maalimine

- [Sissejuhatus](#)
- [Pintsel](#)
- [Lõuend](#)

Mängumootori kasutamine

- [Mängumootori kasutamine](#)

Jõudlus

- [OpenMP](#)

Blenderi sisese mootoriga renderdamine

Sissejuhatus

- [Sissejuhatus](#)

Kaamerad

- [Kaamera](#)
- [Perspektiiv \(koondumispunktid\)](#)
- [Sügavusteravus](#)

Renderdamine

- [Renderduste kuvamine](#)
- [Põhisätted](#)
- [Sakisilumine](#)
- [Animatsioonide renderdamine](#)
- [Eelrenderdus](#)
- [Käsurea kasutamine](#)

Renderduse tulemus

- [Väljund](#)
- [Videoväljund](#)

Efektid ja järeltöötlus

- [Sissejuhatus](#)
- [Renderduskihid](#)
- [Renderduskäigud](#)
- [Servad ja multifilmi renderdus](#)
- [Tempel](#)
- [Liikumishägu](#)

Renderdusjõudlus

- [Võrgurenderdaja](#)

Välised renderdusmootorid

- [Sissejuhatus](#)
- [YafRay](#)

Renderdusmootoriga *Cycles* renderdamine

Sissejuhatus

- [Sissejuhatus](#)
- [Alustamine](#)

Õppetükid

- [Viited](#)

Mootori seletus

- [Kaamera](#)
- [Materjalid](#)
- [- Nihkelaotus](#)
- [- Pind](#)
- [- Maht](#)
- [Tekstuuride muutmine](#)
- [Maailm](#)
- [Valgustid](#)
- [Sõlmed](#)
- [Varjutajad](#)
- [Tekstuurid](#)
- [Lisasõlmed](#)
- [Valguse teekonnad](#)
- [Integreerija](#)
- [Eksperimentaalsed võimalused](#)
- [Graafikaprotsessoriga renderdamine](#)

Sõlmedega komposiitimine

Komposiitimissõlmed

- [Sissejuhatus](#)
- [Sõlmeredaktor](#)
- [Sõlmede juhtelemendid](#)
- [Sõlmede kasutamine](#)
- [Sõlmegrupid](#)

Komposiitimissõlmede tüübid

- [Komposiitmissõlmede tüübid](#)
- [Sisendi sõlmed](#)
- [Väljundi sõlmed](#)
- [Värvi sõlmed](#)
- [Vektorsõlmed](#)
- [Filtrisõlmed](#)
- [Konvertersõlmed](#)
- [Maskimissõlmed](#)
- [Moonutussõlmed](#)

Montaaž

Sissejuhatus

- [Sissejuhatus monteerimisse](#)

Monteerija

- [Monteerija kasutamine](#)
- [Montaaži esitusrežimid](#)

Efektid

- [Sisseehitatud efektid](#)

Heli

- [Helimontaaž](#)

Blenderi laiendamine

Sissejuhatus

Pythoniga skriptimine

- [Blenderis Pythoniga skriptimine](#)
- [Pythoni seadistamine](#)
- [Lisade kasutamine](#)
- [Kiire Pythoni õppetükk](#)
- [Interaktiivne konsool](#)
- [Tekstiredaktor](#)
- [Korduma kippuvad küsimused](#)
- [Viited](#)

Pythoni skriptid

- [Skriptide kataloog](#)

Mängumootor

Sissejuhatus

- [Mängumootori sissejuhatus](#)

Loogika redaktor

- [Kasutamine](#)
- [Mängu omadused](#)

Sensorid

- [Sissejuhatus](#)
- [Sensorite tüübid](#)

Kontrollerid

- [Sissejuhatus](#)
- [Mängu avaldised](#)

Käitured

- [Sissejuhatus](#)
- [Tegevus](#)
- [Objekti muutmine](#)
- [2D-filtrid](#)
- [Mäng](#)
- [Teade](#)
- [Omadus](#)

- [Stseen](#)
- [Olek](#)
- [Nähtavus](#)

Kaamerad

- [Kaamerad Blenderi mängumootoris](#)
- [Stereokaamera](#)
- [Kuppelkaamera](#)

Füüsika

- [Füüsika mootor](#)
- [Objektide tüübid](#)
- [Staatiline](#)
- [Põrge puudub](#)
- [Dünaamiline](#)
- [Jäik keha](#)
- [Pehme keha](#)
- [Peitja](#)
- [Sensor](#)

Pythoni API

- [Bullet physics](#)
- [VideoTexture](#)

Erinevad allikad

- [Võimaluste nimekiri](#)
- [Välised allikad](#)

Sissejuhatus



Blender 2.5 Big Buck Bunny avastseen avatuna

Tere tulemast Blenderisse! Blenderi dokumentatsioon veebis jaotub mitmete ressursside vahel: käesolev käsiraamat, kasutamiseõpetus, juhendmaterjalid, foorumid ja muu. Esimene osa käsiraamatust juhendab sind, kuidas Blender installida, ja kui otsustad lähtekoodid alla laadida, siis ka käivitavat faili kompileerima.

Blenderi esmapilgul ebaharilik kasutajaliides on optimeeritud 3D-graafika loomiseks. Uut kasutajat võib see algul veidi segadusse ajada: selle tugevad küljed selguvad pikemal kasutamisel. Loe käsiraamatut [kasutajaliidese peatükki](#) hoolega, sest see selgitab nii liidese kui ka sinise dokumentatsiooni eripärad.

Mis on Blender?

Blender sai alguse 1993. aasta detsembris. 1994. aasta augustiks oli valmis töötav rakendus, mis võimaldas luua erinevat 2D- ja 3D-sisu. Blenderi funktsionaalsus on väga lai, pakkudes vahendeid modelleerimiseks, tekstuurimiseks, valgustamiseks, animeerimiseks ning ka video järeltöötuseks. Blenderi avatud arhitektuur võimaldab töötada erinevatel operatsioonisüsteemidel, pakub laiendusvõimalusi, nõuab vähe mälu ning pakub ühtset töövoogu. Blender on üks populaarsemaid vabavaralisi 3D-graafika loomise programme.

Blender pakub elukutselisele meediakunstnikule võimaluse 3D-visualiseerimiseks, piltide renderdamiseks ning ka tele- ja filmikvaliteediga videote tootmiseks. Programmi ehitatud reaajas töötav 3D-mootor pakub iseseisva interaktiivse sisu mahamängimise võimalust.

Blenderi algne tootja oli Hollandi firma Not a Number (NaN), kuid praegu toimub programmi tootmine Vaba Tarkvarana GNU GPL-i litsentsi alusel. Programmi arendamist koordineerib Blender Foundation.

Aastatel 2008 kuni 2010 programmeeriti Blender funktsioonide, töövoogu ja kasutajaliidese parandamiseks uuesti ümber. Selle töö tulemusena valmis tarkvaraversioon Blender 2.5 (praegune stabiilne väljalase).

Põhifunktsionaalsus:



Pildi renderdamine ja järeltöötlus

- Üleni integreeritud loomepakett, mis pakub 3D-sisu tootmiseks laia hulka esmavajalikku funktsionaalsust, sealhulgas modelleerimist, UV-laotamist, tekstuurimist, skelettanimatsiooni, osakeste ja muu füüsika simulatsiooni, skriptimist, renderdamist, komponeerimist, järeltöötlust ja mängude loomist;
- Mitmel platvormil töötav OpenGL-i kasutajaliides, mis on ühtne üle kõigi platvormide (Pythoni skriptidega kohandatav), võimeline töötama uuemate Windowsi versioonidega (XP, Vista, 7). Lisaks on toetatud Linux, OS X, FreeBSD, Sun ja mitmed teised operatsioonisüsteemid.
- Kõrge kvaliteediline 3D-arhitektuur, mis pakub kiiret ja tõhusat töövoogu;
- Igat väljalaset laadib alla vähemalt 200 000 kasutaja üle maailma;
- Blenderi kogukonnapoolne tugi küsimuste, vastuste ja konstruktiivse kriitikaga leiate aadressil <http://BlenderArtists.org> ja uudised aadressilt <http://BlenderNation.com> ;
- Blenderi väikest programmifaili on lihtne jagada;

Blenderi viimase versiooni saad laadida [siit](#).

Blenderi ajalugu

Aastal 1988 oli Ton Roosendaal Hollandis animatsiooni- ja arvutustudio "NeoGeo" üks asutajaid. NeoGeo sai õige pea Hollandi suurimaks 3D-animatsiooni tootvaks stuudioks ning oli ühtlasi üks Euroopa suurimatest. Suurfirmade, näiteks rahvusvahelise elektroonikafirma Philips jaoks töötanud NeoGeo sai mitmeid mainekaid auhindu (European Corporate Video Award 1993 ja 1995). NeoGeos oli Ton vastutav nii kunstilise juhtimise kui ka firmasiseses tarkvaraarenduse eest. Pärast põhjalikku kaalumist otsustas Ton, et firmas kasutatav 3D-programmide komplekt on liiga vana ja keeruline hooldada ning et see on tarvis nullist uuesti kirjutada. Ümberkirjutamine algas aastal 1995 ja sellest kasvas hiljem 3D-tarkvarapakett, mida praegu Blenderina tunneme. NeoGeo jätkas Blenderi arendamist ja täiendamist, kuni Ton mõistis, et Blenderit võiks kasutada ka väljaspool NeoGeo kontorit.

Aastal 1998 otsustas Ton asutada NeoGeo tütarfirmana uue firma Not a Number (NaN), mis pidi Blenderit arendama ja turustama. NaN-i tuumaks oli soov luua ja tasuta jagada väikest igal platvormil töötavat 3D-rakendust. Sel hetkel oli tegemist revolutsioonilise mõttega, kuna tol ajal maksid kõik kommertsmodelleerimisprogrammid tuhandeid (Ühendriikide) dollareid. NaN lootis viia professionaalse 3D-modelleerimis- ja animatsiooni tarkvara tavaliste arvutikasutajateni. NaNi ärimudel tugines Blenderi baasil kommertsteenuste ja toodete pakumisele. Aastal 1999 külastas NaN Blenderi propageerimiseks esimest korda SIGGRAPHi konverentsi. Blenderi esimene SIGGRAPH oli väga edukas ja pälvis nii pressilt kui ka külastajatelt palju tähelepanu. Blender oli hitt ja tema tohutu potentsiaal oli kinnitatud!

Eduka SIGGRAPHi järel õnnestus NaNil saada riskikapitalistidelt 4,5 miljoni eurone investeering. Suur rahavoog lubas NaNil kiiresti laieneda. Varsti oli NaNis üle maailma 50 Blenderit arendavat ja turustavat töötajat. Aasta 2000 suvel väljastati versioon 2.0. See versioon tõi endaga 3D-rakendusse integreeritud mängumootori. 2000. aasta lõpus ületas NaNi veebilehel registreeritud kasutajate arv 250 000.

Kahjuks ei olnud NaNi ambitsioonid ja võimalused kooskõlas firma suutlikkuse ega turureaalsusega. Ülelaienemine tähendas, et NaN taasalus 2001. aasta aprillis uue investoriga ja kõrbitud meeskonnaga. Kuue kuu pärast tuli turule NaNi esimene kommertstarkvaratood "Blender Publisher". See toode rihtis interaktiivse veebipõhise 3D turule. Kuna toote müük valmistas pettumuse ja majanduslik kliima oli rusuv, otsustasid uued investorid firma sulgeda. Sulgemine tähendas ka Blenderi arenduse lõppu. Kuigi Blenderil olid sel hetkel mitmed vajakajäämised nagu keeruline sisearhitektuur, pooleliolev funktsionaalsus ja ebastandardne kasutajaliides, oli kogukonna tugi ja Blender Publisheri ostmute entusiasm piisavalt suur, et Ton ei suutnud Blenderit koos firmaga põhja lasta. Kuna firma piisavalt suure meeskonnaga taaskäivitamine polnud otstarbekas, asutas Ton Roosendaal 2002. a märtsis mittetulundusühingu *Blender Foundation*.

Blender Foundation loodi leidmaks viis Blenderi arendamiseks kogukonnapõhise [Vaba Tarkvara](#) projektina. 2002. aasta juuliks suutis Ton NaNi investoreid Blenderi vaba tarkvarana väljaandmise plaanis veenda. Seejärel korraldatud kampaaniaga "Free Blender" püüti koguda 100 000 eurot. Selle raha eest loodeti osta õigused Blenderi lähtekoodile ja NaNi investorite intellektuaalsele omandile, mis oleksid omakorda võimaldanud Blenderi vaba tarkvara kogukonnale väljaandmist. Raha kogumiseks algatasid entusiastlikud vabatahtlikud, kelle hulgas oli ka mitu endist NaNi töötajat, kampaania. Kõigi osapoolte üllatuseks said nad vajalikud 100 000 eurot kokku napi seitsme nädalaga. 13. oktoobril 2002 anti Blender [GNU General Public License \(GPL\)](#) litsentsi all välja. Täna arendavad Blenderit algse looja Ton Roosendaali juhtimisel pühendunud vabatahtlikud üle kogu maailma.

Video: Blender 1.60 kuni 2.50

[\[video link\]](#)

Tähtsamad verstapostid

'Blenderi ajalugu ja eesmärgid:

- 1.00 – Jaanuar 1995: Algab Blenderi arendus NeoGeo stuudios.
- 1.23 – Jaanuar 1998: SGI versioon jõuab veebi: IrisGL.
- 1.30 – Aprill 1998: Linux ja FreeBSD versioonid, OpenGLi ja X-pordid.
- 1.3x – June 1998: asutatakse NaN.
- 1.4x – September 1998: Suni ja Linux Alpha versioonid.
- 1.50 – November 1998: Avaldatakse esimene käsiraamat.
- 1.60 – Aprill 1999: C-key (uus luku taga olev funktsionaalsus, \$95), Windowsi versiooni väljalase.
- 1.6x – Juuni 1999: BeOSi ja PPC versioonide väljalase.
- 1.80 – Juuni 2000: C-key lõpp, Blender taas vabavara.
- 2.00 – August 2000: Interaktiivse 3D ja reaalaaja mootor.
- 2.10 – Detsember 2000: Uus mootor, füüsika ja Python.
- 2.20 – August 2001: Tegellate animeerimise süsteem.
- 2.21 – Oktoober 2001: Lastakse välja Blender Publisher.
- 2.2x – Detsember 2001: Mac OSXi versioon.
- **13 oktoober 2002: Blenderist saab Vaba Tarkvara, 1. Blenderi Konverents (*Blender Conference*).**
- 2.25 – Oktoober 2002: [Blender Publisher](#) avaldatakse tasuta.
- Tuhopuu1 – Oktoober 2002: Luuakse eksperimentaalne programmeerijate mänguväljakuks mõeldud Blenderi koodiharu.
- 2.26 – Veebruar 2003: Esimene päris Vaba Blender.
- 2.27 – Mai 2003: Teine Vaba Blender
- 2.28x – Juuli 2003: Esimene 2.28x-seeria versioon.
- 2.30 – Oktoober 2003: [Eelvaateversioon](#) 2.3x-kasutajaliidestest 2. Blenderi konverentsi käigus.
- 2.31 – Detsember 2003: [uuendamine](#) 2.3x-kasutajaliidesele.
- 2.32 – Jaanuar 2004: Sisseehitatud renderdusmootori [\[1\]](#) põhjalik ümberehitamine.
- 2.33 – Aprill 2004: [Mängumootor on tagasi](#), ambient occlusion (hajusvalgustus), uued tekstuurigeneraatorid.
- 2.34 – August 2004: [Suured täiendused](#): osakeste käitumine, LSCM UV laotamine (*mapping*), YafRay integreerimine, nurkade kaalumine (*creases*) pinnatükeldaja (*subdivision surface*) puhul, värviüleminekud varjutusalgoritmidele, täielik silumine (OSA - Oversampling) ja palju palju muud.
- 2.35 – November 2004: [Järjekordne täiendatud versioon](#): objektide haagid, moonutamine kõveratega and kõveratega koondamine (curve tapers), osakeste duplitseerijad ja palju muud.
- 2.36 – Detsember 2004: [Stabiliseeriv versioon](#), palju telgitagust tööd, pinnanormaalide ja nihkelaotuse (*displacement map*) parandamine.

- 2.37 – Juuni 2005: [Suur hüpe](#): teisendustööriistad ja ja vidinad (*widget*), pehmed kehad (*softbody*), jõuväljad, põrked, pindade tükeldamine, läbipaistvad varjud ja renderdamine mitmes lõimes.
- 2.40 – Detsember 2005: [Veeigi suurem hüpe](#): täielik skeletisüsteemi ümbertöötamine, võtmevormid (*shape key*), osakeste abil loodud karvad, vedelikud ja jäigad kehad.
- 2.41 – Jaanuar 2006: [Palju parandusi](#) ja veidi mängumootori funktsionaalsust.
- 2.42 – Juuli 2006: [Sõlmede \(node\) väljalase](#). Üle [50 arendaja](#) töid programmi sõlmed, massiivi töötleja (*array modifier*), vektorudu (*vector blur*), uue füüsikamootori, renderdamise, huulte sünkroniseerimise ja palju muid funktsioone. See väljalase järgnes [Project Orange'i](#) lõppemisele.
- 2.43 – Veebruar 2007: [Mitu väljalase](#): mitmelahutuslik võre, mitmekihilised UV-tekstuurid, mitmekihilised pildid ja mitmekordne renderdamine, mitmekordne eelrenderdus (*baking*), voolimisrežiim, topoloogia muutmine (*retopo*), mitmekordne lisa-Matte (taustapilt), moonutus ja filtrisõlmed, modelleerimise ja animeerimise parandamine, parem erinevate pintslitega maalimine, vedelikuosakesed, proksiobjektid, montaažimootori ümberkirjutus ja tekstuurimine järeltöötuses. Phuuh! Ahjaa, veebilehe taaskirjutamine. Ja muide, mitmelõimeline renderdamine mitmetuumalistele protsessoritele on ikka alles. Verse'i abil on võimalik ka mitu kasutajat, mis võimaldab mitmel kunstnikul korraga sama stseeni kallal toimetada. Lõpetuseks pakuvad renderdusfarmid ikka veel hajusrenderdamist.
- 2.44 – Mai 2007: [SSS-väljalase](#): suured uudised, lisaks kahele uuele töötlejale ja 64-bitise operatsioonisüsteemi toe taaselustamisele tekkis ka pinna sisehajumine, mis simuleerib kudede ja pehmete objektide sees hajuvat valgust.
- 2.45 – September 2007: [Veaparandusväljalase](#): olulised veaparandused, parandatud ka kiirust.
- 2.46 – Mai 2008: [Projekti Peach versioon](#) oli 70 arendaja ponnistuste viili. Täiendati programmi tuuma, parandati juukseid ja karvu, uus osakeste süsteem, parandati piltide lehitsemist, riidesimulatsioone, loodi iseseisev füüsikapuhver, peegelduste renderdamise parandused, *Ambient Occlusion* (hajusvalgustus) ja eelrenderdamine; töötleja võre moonutamiseks (*mesh deform*) lihaste ja muu samase puhul. Tekkisid paremad animatsioonivõimalused seoses skeletitööriistade, joonistamise, katmise, piirajate ja värvilise Tegevuste Redaktoriga (*Action Editor*). Lisandus ka palju muud. Väljalase järgnes projektile [Peach](#).
- 2.47 – August 2008: [Veaparandusversioon](#).
- 2.48 – Oktoober 2008: [Projekti Apricot väljalase](#): edev GLSLi varjutaja, valguse ja mängumootori parandamine, haaramine ('snap'), *taeva simulaator*, *sobitamise töötleja* (shrinkwrap), *Pythoni toimet*i parandused.
- 2.49 – Juuni 2009: [Eel-ümber-faktoreerimise väljalase](#) lisas programmi tuumale ja mängumootorile olulisi täiendusi. Tuumale täiendamine sisaldas sõlmedel põhinevaid tekstuure, skeleti skitseerimist (nimega Etch-a-Ton), võre mahutehete (*boolean*) parandamine, JPEG2000 tugi, projektsioonmaalimine piltide mudelitele kandmiseks ja oluline Pythoni skriptide kataloog. Mängumootorit täiendati videotekstuuridega, mis lubavad mängus filme näidata(!), täiendusi füüsikamootorile Bullet, renderdamist kuplisse (kalasilm). Mängumootori liidest täiendati uute funktsioonidega.
- 2.5x – Alates 2009 kuni August 2011. Selle seeria [väljalasked](#) sisaldavad 4 eelversiooni (alates Alpha0 - Novembris 2009 - kuni Betani Juulis 2010) ja kolm stabiilset versiooni (alates 2.57 - Aprillis 2011 - kuni 2.59 - Augustis 2011). Tegemist on ühe Blenderi kõige suurema [arendusprojektiga](#), mis kirjutas suure osa programmist uuesti, lisades uut funktsionaalsust, uue sisemise aknahalduri ja sündmuste/tööriistade/andmete haldamise süsteemi, uue Pythoni API... Projekti viimane versioon oli Blender 2.59 (2011. a augustis).

Vaba Tarkvara ja GPL-i litsents



Esimene asi, mida ilmselt väljendi "vaba tarkvara" kuulmisel mõeldakse, on "tasuta". Kuigi see on üldjuhul ka täiesti õige mõtteviis, on Free Software Foundationi (kes on GNU projekti algatajad ja GNU Üldise Avaliku Litsentsi loojad) tõlgenduses "vaba tarkvara" konkreetselt "vaba nagu vabadus" ja mitte "tasuta" (viimasele viidatakse inglise keeleruumis "*free as in beer*" - "vaba nagu tasuta õlu"). Vaba tarkvara on selles tähenduses tarkvara, mille kopeerimisel, kasutamisel, muutmisel ja jagamisel on piiramatut vabadust. Püüdkem võrrelda seda enamiku kommertstarkvara litsentsidega, mis lubavad sul tarkvara laadida vaid ühte arvutisse, keelavad seda kopeerida ja sa ei näe kunagi programmi lähtekoodi. Vaba tarkvara annab lõppkasutajale meeletu vabaduse. Kuna lähtekood on kõigile kättesaadav, on võimalus, et keegi vigu märkab ja neid parandab, samuti oluliselt suurem.

Kui programm on GNU Üldise Litsentsi (GPL-i) all:

- on sul õigus programmi igaks otstarbeks kohandada;
- on sul õigus programmi muuta ja õigus lähtekoodiga tutvuda;
- on sul õigus programmi kopeerida ja edasi anda;
- on sul õigus programmi täiendada ja oma versioone välja anda.

Nende õiguste vastu pead võtma mõned kohustused programmi edasijagamisel; need kohustused on sellesama "piiramatus" säilitamiseks nii sinu kui ka teiste jaoks.

- Pead andma GPL-i koopia programmiga kaasa, et vastuvõtja oleks teadlik oma litsentsiga kaasnevatest õigustest ja kohustustest.
- Pead lähtekoodi kaasa andma või selle tasuta kättesaadavaks tegema.
- Kui lähtekoodi modifitseerid ja modifitseeritud versiooni edasi jagad, pead oma muudatused samuti GPL-i all litsenseerima ning nende lähtekoodi samuti tasuta kättesaadavaks tegema. (Sa ei tohi kasutada GPL-i all olevat koodi suletud programmi loomiseks.)
- Sa ei tohi programmi litsentsi rohkem kitsendada kui GPL-i litsentsiga. (Sa ei tohi muuta GPL-i all olevat programmi suletud tooteks.)

GPL-i kohta lisainfo lugemiseks võid külastada [GNU Projekti veebilehte](http://amrc.altervista.org).

Kasutajatugi: Blenderi kogukond

Blender oli isegi suletud lähtekoodiga perioodil tasuta saadaval; see kasvatas Blenderi kasutajate arvu arvestatavalt. Aastast 1998 kasvanud aktiivsete kasutajate kogukond on paisunud suureks ja stabiliseerunud.

Kogukond oli oluline kriitilisel hetkel Blenderit vabaks ostes, kui programm 2002 aasta oktoobris GNU GPLi alla viidi.

Eesti kogukond

blender.org.ee alustas tegevust 2011 aasta suvel.

Rahvusvaheline kogukond

Rahvusvaheline kogukond on praegu arvestatava kattuvusega jagunenud kahe suurema veebilehe vahel:



Arenduskogukond, [Blender Foundationi kodulehel](#). Siin on arendusprojektide kodu, Funktsionaalsuse ja Dokumentatsiooni foorumid, CVSi koodihoidla, mis Blenderi lähtekoodi hoiab, kogu dokumentatsioon ja foorumid avalikuks aruteluks. Siit leiab arendajad, kes Blenderit kirjutavad, Pythoni skriptijad, dokumenteerijad ja kõik ülejäänud, kes Blenderi arendamisel kaasa löövad.

Mine aadressile <http://www.blender.org>



Kasutajate kogukond, sõltumatul leheküljel [BlenderArtists](#). Siin kogunevad oma loomingu näitamiseks, tagasiside saamiseks ja programmeerimise küsimuste esitamiseks Blenderit kasutavad kunstnikud, mängude kirjutajad ja fännid. Siit leiab ka Blenderi õppematerjale ja teadmusbasi.

Mine aadressile <http://www.BlenderArtists.org>

Blenderi materjale leiab ka mujalt peale nende kahe veebilehe. Ülemaailmne kogukond on loonud suure hulga sõltumatuid lehti, nii kohalikes keeltes kui ka spetsiifilisematele teemadele keskendudes. Blenderiteemaliste materjalide nimekirjad leiad ülalmainitud lehekülgedelt.

IRC jututoad

Kiire tagasiside hõlbustamiseks on kolm alati avatud ingliskeelset IRC vestluskanalit, mille leiad aadressilt irc.freenode.net. Liitu oma lemmikliendiga:

- [#blenderchat](#) Vabaks aruteluks Blenderi teemal;
- [#blenderqa](#) Blenderiteemaliste küsimuste esitamiseks;
- [#gameblender](#) Blenderi mängumootoriga seotud küsimuste esitamiseks.

Arendajatele on mõeldud:

- [#blendercoders](#) arendajatele küsimuste ja arendusega seotud probleemide jaoks, siin toimub igal pühapäeval kell 6 (Eesti aja järgi) arenduskoosolek;
- [#blenderpython](#) Pythoni liidese ja skriptimise teemal pärimiseks;
- [#blenderwiki](#) wikiga seotud küsimuste tarvis

Kes Blenderit kasutavad?

Uusi Blenderi väljalaskeid laaditakse esimese 10 päeva jooksul üle miljoni korra. See arv hõlmab kõiki platvorme (Windows, Linux ja MacOS), aga ei sisalda teisi levitajaid, kes loomulikult tohivad seda samuti piiramatult teha. Hindame, et kasutajaid on üle kahe miljoni. Käsiraamat on koostatud paljude andekate Blenderi kasutajate hüvanguks:

- hobi korras või õpilasena arvutigraafika ja 3D-maailma uudistaja
- 2D kunstnik, kes teeb üksikuid pilte või postreid või soovib üht pilti edasi töödelda
- 2D kunstnik või meeskond, kes animeerib telereklaame või lühifilme (nagu "The Magic of Amelia")
- 3D kunstnik, kes üksi või koostöös lühikest reaalselt videot sisaldavat arvutanimatsiooni teeb (näiteks "Suburban Plight").
- 3D meeskond, kes teeb animeeritud (100% arvutigraafikas) filmi (nagu "[Elephant's Dream](#)", "[Plumiferos](#)", "[Big Buck Bunny](#)" või "[Sintel](#)").
- 3D meeskond, kes teeb filmi, mis sisaldab arvutiefekte.

Blenderit kasutavad nii teismelised kui ka vanad olid ja kogukond on uute ja kogenud kunstnike vahel üsna ühtlaselt jaotunud; on juhukasutajaid ning neid, kes programmiga raha teenivad. Võime filme loovad 2D ja 3D meeskonnad omakorda ülesannete kaupa ära sorteerida. Blenderit kasutaks neist:

- Režissöör - määrab iga stseeni sisu ja selles toimuva tegevuse (animatsiooni). Määrab stseenis toimuvad võtted.
- Modelleerija - tehistsõelise looja. Spetsialiseeruda võib karakteritele, rekvisiitidele ja maastikule/lavale.
- Operaator: määrab kaamera ja selle liikumise, vajadusel filmib, renderdab kaadrid.
- Tekstuurija - maalib võttekoha, näitlejad ja kõik, mis liigub. Kui miski ei liigu, maalivad nad sellegi.
- Animatsioon ja taageldamine - paneb skelettide abil asju ringi kargama. (Taageldamine on purjelaeva purjenõõride (taglase) ülesseadmine ja otsetõlge inglise sõnast *rigging*).
- Valgustaja ja värviekspert - valgustab lava ja võttekohad, korrigeerib värve, et need head välja näeksid, lisab materjalidele, stseenidele ja tekstuuridele tolm ja sodi.
- Eriotstarbeline talent - vedelikud, liikumise salvestamine (Motion Capture), kangasimulatsioon, tolm, kõnts, tuli, plahvatused ja muu palju lõbu pakkuv
- Monteerija - võtab operaatorilt toomaterjali ja lõigub selle üheks nauditavaks taieseks kokku. Lõikab ebavajaliku välja.

Käsiraamatu info

See käsiraamat on koostatud Mediawiki abil vabatahtlike [autorite](#) poolt üle maailma. Käsiraamat täieneb pidevalt, sa loed selle eestikeelset versiooni. Rahvusvaheliste kasutajate jaoks on sellest tõlked erinevatesse keeltesse. Need baseeruvad üldjuhul ingliskeelsele versioonil. Käsiraamat ei ole reeglina ajakohane, kuna üle 50 vabatahtliku arendaja teeb pidevalt ponnistusi, lisades kogu aeg uusi funktsioone. Ometi on siinse ettevõtmise eesmärk pakkuda sellele imelisele pakatile parimat võimalikku dokumentatsiooni.

Sinu aitamiseks parimal ja tõhusaimal võimalikul viisil järgime teemades 3D kunstnike loomeprotsessi tavapärase kulgu, peatudes sobilikel kohtadel selgitamaks, kuidas sel tundmatul maastikul, petlikult keerulises tarkvarapakettis, paremini orienteeruda. Kui loed käsiraamatut järjest, tutvud töö etappidega samas järjekorras, milles Blenderit reeglina õpitakse ja animeerimiseks kasutatakse.

1. Blenderiga tutvumine = Sissejuhatus, 3D-ruumis navigeerimine, stseenide haldamine
2. Modelleerimine = Modelleerimine, töötlejad (Modifiers)
3. Valgustamine
4. Varjutamine = Materjalid, tekstuurid, maalimine, maailm ja taustad
5. Animatsioon = Põhitõed, tegelased, edasijõudnutele, efektid ja füüsikasimulatsioon
6. Renderdamine = Renderdamine, komponeerimine, videoradade järjestamine
7. Blenderi täiendamine

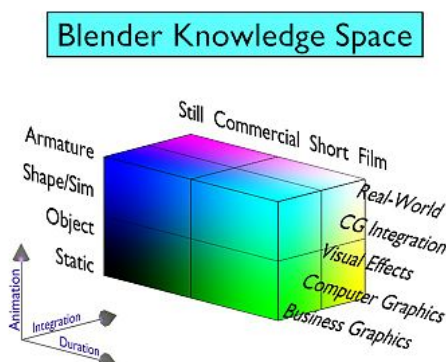
Sihtrühm

Käsiraamat püüab teenindada võimalikult laia publikut, alustades küsimusest: "Ma tahan "midagi teha"; kuidas ma selleks Blenderit kasutada saaksin?" ja lõpetades küsimusega "mis on viimane muutus võre digisavi režiimis modelleerimisel?"

See käsiraamat on ülemaailmse vabatahtliku koostöö ja aja panustamise tulemus. Kuigi uute võtmefunktsioonide rakendamise ja nende dokumenteerimise vahel võib kuluda aega, püüame seda käsiraamatut võimalikult ajakohasena hoida. Püüame keskenduda peaaesjalikult sellele, mida sina lõppkasutajana teadma peaksid, ning mitte kalduda teemast liialt kõrvale, hakates näiteks elu mõtte üle arutlema.

On olemas ka [teisi Blenderi wikiraamatuid](#), mis süüvivad põhjalikumalt teistesse teemadesse ja kirjeldavad Blenderit teistest vaatenurkadest lähtuvalt, nagu näiteks **Tutorialid** (õppejuhendid) "**Reference Manual**" (kasutajaliidese käsiraamat), tarkvara ise ja tema skriptimiskeel. Seega, kui antud käsiraamat mõnele küsimusele ei vasta, võid leida vastuseid teistest [Blenderi wikiraamatutest](#).

Arvutigraafika ja Blenderi õppimine



Blenderi ja arvutigraafika tundmaõppimine on kaks erinevat teemat. On kaks eri asja õppida selgeks, mis on arvutis tehtud mudel ja kuidas üht sellist Blenderis modelleerida. Ka hea valgustustehnika õppimine ja siis Blenderi võimalike lambitüüpidega tutvumine on kaks eri teemat. Algsed, kontseptuaalsed, teadmised on võimalik saada koolitustelt, meediakunsti ülikoolikursustelt, raamatukogus või poes leiduvatest kunsti ja arvutigraafika teemal kirjutatud raamatutest ning katse-eksitusega meetodil. Isegi kui mõni raamat või artikkel kasutab töövahendina mõnd muud tarkvarapaketti (nagu Max või Maya), on allikas väärtuslik just teoreetilise baasi edastamisel.

Kui teoreetiline külg on mõistetav ja selge, on Blenderi (või suvalise muu 3D-rakenduse) õppimine lihtne. Mõlema samaaegne õppimine on keeruline, sest tuleb tegeleda korraga kahe probleemiga. Eelneva kirjutise eesmärk on selgitada sulle selle probleemi ja dilemma olemust ning näidata, kuidas see käsiraamat mõlemat teemat ühe wikiraamatuna käsitleda püüab. Teoreetilised teadmised on enamasti lahti kirjutatud teema või peatüki esimestes lõikudes. Need avavad teema ja pakuvad ülesande lahendamise töövoos (töökäigu) või meetodi. Ülejäänud teema või peatüki osa kirjeldab Blenderi konkreetseid võimalusi ja funktsioone. Käsiraamatu maht ei võimalda pakkuda täielikku teoreetilist tausta — see tuleb juurde raamatutest, ajakirjadest, õppejuhenditest ja mõnikord ka elu jooksul tehtud ponnistusest. Blenderit kasutades on võimalik luua täispikk film; ometi ei tee see käsiraamat Blenderiga relvastatult uueks Steven Spielbergiks!

Väga kõrgel tasemel vaadatuna on Blenderi kasutamist võimalik näha oskusena saavutada kujundlikkus kolmel teljel:

1. Seostamine - arvutigraafika renderdamine, reaalse maailma videote töötlemine või nende kahe omavaheline segamine (arvutigraafika + eriefektid)
2. Animatsioon - poseerimine ja asjade kuju muutmine, kas käsitsi või simulatsiooni abil
3. Kestus - pilt, videoklipp, minutine reklaam, kümneniutine lühifilm, täispikkuses film.

Oskused, nagu 3D-ruumis ringi liikumine, modelleerimine, valgustamine, varjutamine, kompositsiooni kasutamine, on samas vajalikud selle ruumi igas punktis. Vilumus mingis oskuses muudab su produktiivseks. Ka Blenderi tööriistad paigutuvad sellesse ruumi. Näiteks on videoribade järjestajal (*video sequence editor (VSE)*) vähe seost animeerimisoskusega, kuid Kestuse ja Seostamise teljel on see väga vajalik. Oskuste õppimise ja omavahelise seostamise vaatenurgast on ehk huvitav tõik, et animeerimisel kasutatav kõver, mida nimetatakse IPO-kõveraks (*IPO-curve*) on samas kasutusel efekti radade animeerimisel eelmainitud videote järjestajas (*Video Sequence Editor*).

Selle ruumi nurkades/ristumispunktides on kohad, kuhu paigutuvad kasutajate huvid, nii-öelda lõppeesmärgid. Näiteks on palju andekaid kunstnikke, kes teevad staatilist arvutigraafikat. Tony Mulleni raamat "*Introducing Character Animation with Blender*" kirjeldab, kuidas luua arvutimudeleid skelettide (*armature*) ja vormide (*shape*) abil painutades üheminutilist animatsiooni. Blenderi vedelike kasutamises televisiooni vaheklipi/reklaami tegemisel ristuvad omavahel *Shape* (vorm) /Simuleerimine, liidetud video ja arvutigraafika. Elephants Dream ja Big Buck Bunny on mulilike Skelettanimatsiooni-Arvutigraafika-Lühifilmi ruumis. Seega sõltub see, millistest tööriistadest ja teemadest sa Blenderi juures huvitud, eeskätt sellest, mida sa teha soovid.

Neljas telg võiks olla mängudisain, sest see võtab kogu eelneva ning mähib selle ümber mängude loomiseks vajalikud teadmised. Mäng ei sisalda endas vaid minutipikkust demo, vaid ka tegelikku sisu, loo programmilist esitust jne – see peaks ka selgitama, miks on mängu tegemine sedavõrd keeruline; enne mängu tegemist tuleb kogu eelnevalt mainitust aru saada. Seega ei sisalda antud käsiraamat mängumootori kasutamist, sest sellest räägib üks täiesti teine wikiraamat.

Programmi paigaldamine

Blender 2.59 on Blender Foundationi kodulehel (<http://www.blender.org/>) saadaval nii töötava kompileeritud programmi kui ka lähtekoodina. Blenderi versiooni 2.59 laadimiseks vali [Blenderi kodulehe](#) parempoolsest navigeerimismenüüst *Blender 2.59*.

Kui kasutad võrgumanuaali wikis, on kõige otstarbekam kasutada Blenderi kõige uuemat versiooni, mille Blender Foundationi kodulehelt laadida saad (kuigi kõik uuendused ei kajastu alati kohe käsiraamatus). Kui kasutad käsiraamatu trükiversiooni, soovitage kasutada seda Blenderi versiooni, mis CD-ga kaasneb. Kui edasises tekstis esineb väljend "alla laadimine", peaksid trükiversiooni omanikud faili CD-lt võtma.

Rakenduse paigaldamine ja allalaadimine

Binaarkujul pakutakse rakendust järgmistele operatsioonisüsteemiperekondadele:

- [Windows](#)
- [Linux](#)
- [MacOSX](#)
- [FreeBSD, Solaris](#)

On ka mõned mitteametlikud väljalasked teistele operatsioonisüsteemidele, aga neid [Blender Foundation](#) ei toeta ning nendega tekkivatest vigadest palume teatada vastavate versioonide haldajatele.

Macintoshi operatsioonisüsteemi programme pakutakse kahele erinevale arhitektuurile (x86 Inteli ja AMD protsessoritele ning PowerPCle) ning valida saab dünaamiliselt ja staatiliselt lingitud teekidega versioonide vahel.

Paigaldaja loob sinu arvutis faile ja kaustu kahte erinevasse kohta: üks kataloogikomplekt on Blenderi enda jaoks, teine komplekt kasutajaandmetele. Nende loomiseks peavad sul olema arvutis administraatorõigused. Kaustad on:

- .blender - konfiguratsiooniinfo (enamasti sinu emakeeles)
- blendcache_*.B - ajutine ruum füüsikasimulatsioonide infole (pehmed kehad, kangas, vedelikud)
- plugins - lisafunktsionaalsus ja järjestaja
- tmp - ajutine väljund, vaherenderdused

Toetatud riistvara

Kui Blenderit jooksutada 64-bitise arhitektuuriga 64-bitise Unixi operatsioonisüsteemiga, puudub adresseeritava mälu puhul 2 GB piirang.

Blender toetab ka mitmetuumalisi protsessoreid, näiteks Inteli Core-Duo ja AMD X2 kiipe "lõimede" seadistuse (Threads) abil, millega saab määrata renderdamisel tuumade paralleelkasutuse. Kui antud seadistus on automaatne (Auto-detect), kasutab süsteem kõiki saadaval olevaid tuumasid, samas kui Fixed lubab renderdamisel kasutatavate tuumade arvu ise määrata.

Blender toetab laia valikut survetundliku pliiatsiga digilaudu. Digilaudade tugi on kõigil põhilistel operatsioonisüsteemidel nagu näiteks OS X, Windows ja Linuxid.

Manuaali [Renderdamise](#) peatükk pakub renderdamisaja kahandamise kohta lisainfot.

Arendusplatvormid

(Alljärgnev on kaasatud Blenderi arendusinfost ja on ajakohasuse huvides tõlkimata) This is the list of systems in use and supported by active Blender developers:

Name	OS	CPU	Graphics card
Andrea Weikert	Windows XP 32	AMD Athlon 64 X2	Nvidia Quadro FX1500
Andrea Weikert	Windows XP 32	Intel P4	ATI Radeon 9000
Andrea Weikert	Linux 32	AMD Athlon 64 X2	Nvidia Quadro FX1500
Antony Riakiotakis	Ubuntu 14.04	Intel Core i5	NVidia Geforce GT 540M
Antony Riakiotakis	Windows 7 64	Intel Core i5	NVidia Geforce GT 540M
Bastien Montagne	Debian Testing 64	Intel Core i7 Q4700m	NVidia Geforce GTX 850M
Benoit Bolsee	Windows XP 32	AMD Athlon XP	ATI Radeon 9200
Brecht van Lommel	Linux 64	Intel Core 2 Duo	NVidia GeForce 460 GTX
Brecht van Lommel	OS X 10.6	Intel Core 2 Duo	NVidia GeForce 9600M GT
Brecht van Lommel	Windows 7 64	Intel Core 2 Duo	NVidia GeForce 460 GTX
Campbell Barton	Linux 64	AMD-FX 6-Core	Nvidia GeForce GTS 450
Daniel Genrich	Windows Vista 64	Intel Core 2 Duo	NVidia GeForce 8500 GT
Diego Borghetti	Linux 64	Intel Core i5	Nvidia GeForce GTX 480
Diego Borghetti	Linux 64	Intel Core i7	Nvidia GeForce GTX 460M
Dustin Martin	OSX 10.5	Dual Quad Intel	Nvidia Geforce 8800 GT
Howard Trickey	Ubuntu 12.04 64	Intel Xeon E5-1650	NVidia Quadro 600
Howard Trickey	Windows 7 64	Intel Core i7	NVidia GeForce GTX 460
Howard Trickey	OSX 10.8.2	Intel Core Duo	NVidia GeForce 9400M

Jens Verwiebe	OSX 10.6/7/8/9	Intel Xeon 6-core@ 3.33	ATI 5870/7970
Jeroen Bakker	Latest Ubuntu 64bit	Dell m4300 Intel Core 2 Duo 2.0Ghz	Nvidia Quadro FX360M
Joshua Leung	Windows Vista 32	Intel Core2 Duo	Nvidia GeForce Go 7600
Julian Eisel	Linux 64	AMD Phenom II X4	Nvidia GeForce GTX 570
Julian Eisel	Linux 64	Intel Core i7	Nvidia GeForce GT 645M
Ken Hughes	Linux 32	Intel Core Duo	Nvidia GeForce GO 7500
Ken Hughes	Linux 64	AMD Athlon 64 X2	Nvidia GeForce 6600
Kent Mein	Linux 64	Intel Core Duo	Nvidia Quadro FX 1400
Kent Mein	SunOS 5.8	Sun Blade 150	ATI PGX
Matt Ebb	OSX 10.5	Dual Core Intel MBP	nVidia 8600M
Michael Fox	Linux 32	Celeron	Nvidia GeForce 6200
Nathan Letwory	Windows 7 Ultimate 64	AMD Turion X2 Mobile RM-74	ATI HD 4650
Nathan Letwory	Windows 7 Ultimate 64	AMD Athlon II X4 620	2x HIS ATI HD 5550 /w four monitors
Nicholas Bishop	Fedora 18 64bit	Intel Core i7 @ 2.93GHz	AMD Radeon HD 6950 (Gallium drivers, currently at OpenGL 2.1)
Nicholas Bishop	Ubuntu 12.10 64bit	Intel Core i5	ATI Mobility Radeon 5650 (Gallium drivers)
Raul Fernandez Hernandez	Linux 32	Pentium D 945	ATI X1550
Robin Allen	Linux 32	Intel Centrino duo	NVidia GeForce go 7600
Robin Allen	Windows XP 32	Intel Centrino duo	NVidia GeForce go 7600
Sergej Reich	Arch Linux 64bit	Intel Core2 Quad @ 2.83GHz	Nvidia GeForce GTX 285
Sergej Reich	Arch Linux 64bit	Intel Core i3 @ 2.10GHz	Intel Sandybridge Mobile
Sergey Sharybin	Debian Wheezy 64bit	Intel Core i7 920 2.6Ghz	Nvidia GeForce GTX 560Ti + GeForce GT 620
Sergey Sharybin	Debian Wheezy 64bit	Intel Core i5 2.4GHz	Intel Sandy Bridge + Nvidia GT 520M
Thomas Dinges	Windows 7 x64	Intel Core i5	Intel HD 2500
Thomas Dinges	Windows 7 x64	Intel Core i7	NVidia GeForce 540M + Intel HD 3000
Timothy Baldrige	SGI Irix 6.5 (mipspro)	8 x R16000	(headless)
Timothy Baldrige	SGI Irix 6.5 (mipspro)	2 x R10000	
Tamito Kajiyama	Windows Vista 64bit	Intel Core2 Duo	Nvidia Quadro FX 770M
Ton Roosendaal	OSX 10.7	iMac Intel Core i7	AMD Radeon HD 6970M
Ton Roosendaal	OSX 10.8	MacBook Pro i7 "Retina"	NVidia GT 650M + Intel HD 4000

Lähtekoodi kompileerimine

Erinevate toetatud operatsioonisüsteemide jaoks käib programmi kompileerimine nelja kompileerimissüsteemiga. Wikileht http://wiki.blender.org/index.php/Dev:Doc/Building_Blender Building Blender] pakub sinu süsteemi jaoks kohandatud kompileerimisjuhiseid.

Pluginate kompileerimine

Pluginaid on programmi tööajal sellega ühenduvad funktsionaalsust täiendavad programmid, mis aitavad luua tekstuure või töödelda montaaži kujutisi. Vaata [lisainfot siit teemast](#).

Pythoni skriptimiskeel

[Python](#) on üldotstarbeline skriptimiskeel, millega laiendatakse Blenderi funktsionaalsust, ilma et peaks [Blenderi allalaadimislehel](#) saadud kompileeritud rakendust muutma. Praegused Blenderi versioonid tulevad sobiva Pythoniga ning seetõttu pole sisemiste ja välimiste Pythoni skriptide jaoks täieliku Pythoni paigaldamine tarbilik.

Täielik paigaldus on kasutajatele, kes soovivad [omi skripte kirjutada](#), [enda Blenderit kompileerida](#) või mõnd ebaharilikku lahendust luua. Kasutajad, kes soovivad täielikku Pythoni funktsionaalsust, peaksid lugema paigaldusjuhiseid [Pythoni](#) kodulehelt.

Paigaldamine Windowsis

Laadi alla

Viimase stabiilse Blenderi versiooni Windowsile saab [Blenderi allalaadimise lehelt](#).

Versioon

Blenderi Windowsi versioone on saadaval 32-bitise ja 64-bitise arhitektuuri jaoks. 32-bitise Windowsiga kasutajad peavad laadima 32-bitise versiooni. 64-bitise Windowsi versiooniga kasutajad võivad valida nii 32- kui ka 64-bitise Blenderi, kuigi 64-bitine versioon pakub märgatavalt suuremat jõudlust, eriti süsteemidel, millel on palju mälu.

Kontrollimaks, kas sul on 32-bitine või 64-bitine Windows, vaata C:\ kettale. Kui sul on kataloog nimega "Program Files (x86)", on sul 64-bitine Windows. Kataloog "Program Files (x86)" hoiab 32-bitiseid programme.

Paigaldamine

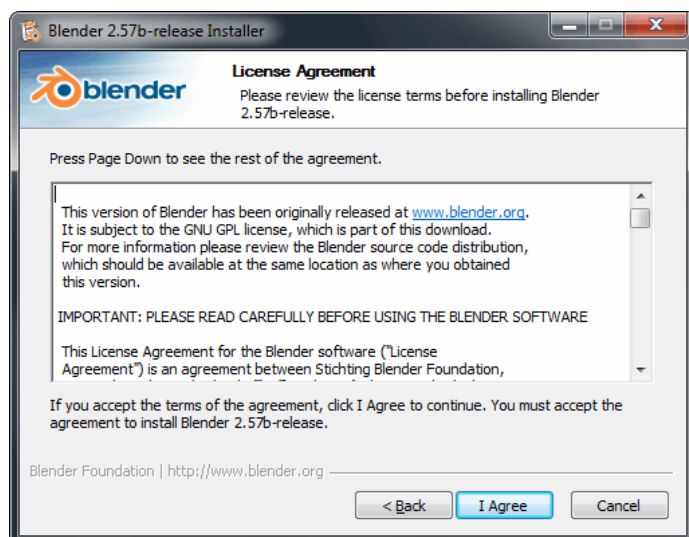
Kui oled programmi alla laadinud, mine kataloogi "Downloads" ning tee paigalduse alustamiseks topeltklõps Blenderi installfailil. Ära unusta, et Blenderi paigaldamine vajab administraatori õigusi.

Tervitustekst



Paigaldamine algab tervitustekstiga. Jätkamiseks vajuta "Next".

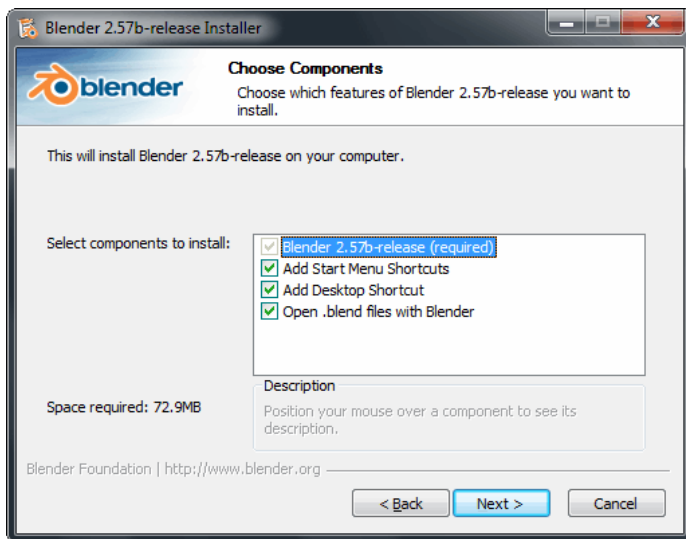
Litsentsitingimused



Pead enne paigalduse jätkamist litsentsiga nõustuma.

Paigaldusvalikud

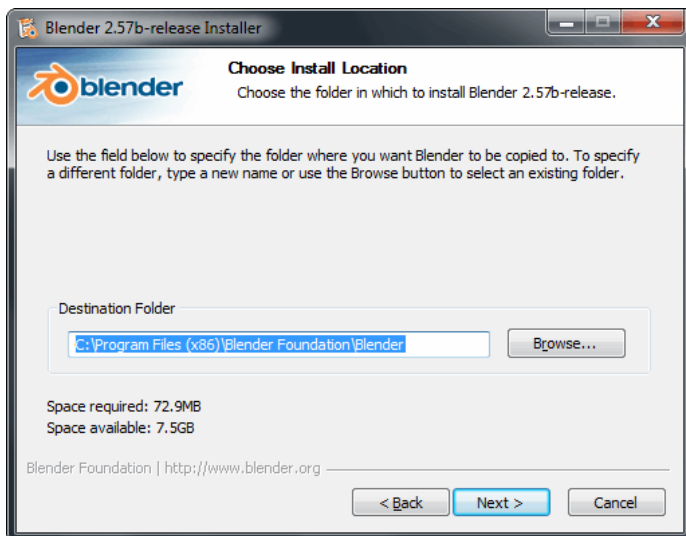
Programmivalikud



Tee soovitud valikud.

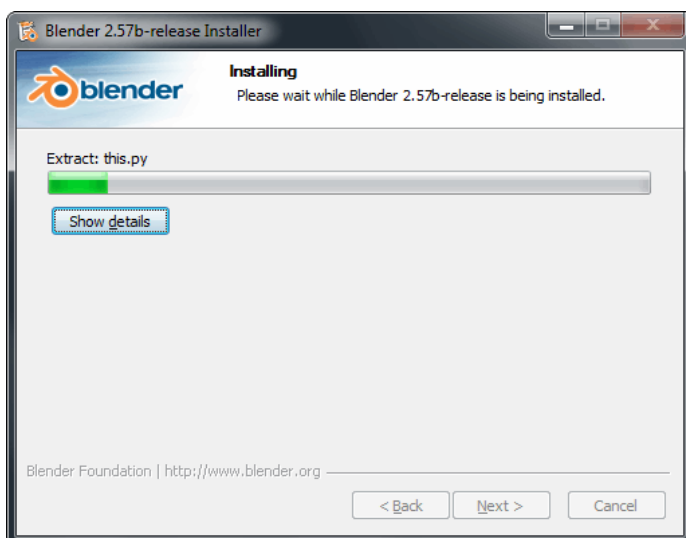
- *Add Start Menu Shortcuts* - lisab programmi menüüsse Start
- *Add Desktop Shortcut* - lisab programmi ikooni töölaualle
- *Open .blend files with Blender* - avab .blend failid Blenderiga

Asukoht



Siin saad valida, millisesse kataloogi Blender paigaldatakse.

Paigaldamine



Paigaldatud



Ja ongi valmis! Värske Blender 2.5 paigaldus. Pärast seda, kui oled valinud, kas soovid Blenderit pärast paigalduse lõppu jooksutada, klõpsa nupul *Finish*.

Teisaldatav paigaldus

Kui soovid Blenderit USB-pulgalt jooksutada, et ta sul alati kaasas oleks, laadi kodulehelt alla .zip-versioon ja paki see USB-pulgale lahti. Võib juhtuda, et sa ei taha mälupulgale animatsioone ja muud ajutist väljundit salvestada, sest sage kirjutamine lühendab pulga eluiga.

Tee Blenderi kataloogi alamkaust nimega "config", mis hoiab seadistusfaile ja paigaldatud lisaasid. Nii loetakse ja kirjutatakse kõiki seadistusi USB-kettalt ja mitte sellest arvutist, kus Blender jookseb.

Linux

Kompileeritud kujul saab Linux'i operatsioonisüsteemile programmi kahele erinevale arhitektuurile x86 (Intel ja AMD protsessorid) ja PowerPC ning valik on staatiliselt ja dünaamiliselt lingitud teekidega versioonide vahel.

Dünaamilise ja staatilise linkimise erinevus on oluline. Staatiliselt kompileeritud programmis tulevad kompileeritud [OpenGL](#) teegid kaasa. See tähendab, et Blender jookseb sinu süsteemis ilma riistvaralise kiirenduseta. Staatilist versiooni kasuta ainult siis, kui dünaamiline ei tööta!

Distroga kaasatulev Blender

Linux'i levitajad kaasavad Blenderi tihti toetatud programmide komplekti. See tähendab, et Blenderi saab paigaldada vastavale distrole sobiva paketi haldusvahendiga; Fedora, Mandriva, Open SUSE, Ubuntu ja Debian võimaldavad Blenderit nii paigaldada — see nimekiri ei ole ammendav.

Allpool toodud juhised selgitavad, kuidas paigaldada viimast Blenderi versiooni, sest paketi haldusvahendites olev versioon on tihti aegunud.

Kiire install

Lae alla fail `blender-2.5x-beta-linux-glibc27-ARCH.tar.bz2` 2.5 beta lehelt (aadressil <http://www.blender.org/download/get-256-beta/#linux386>). Asenda ARCH oma arvuti arhitektuuritähisega — `kasi686` või `x86_64`. Vali see, mis kattub sinu süsteemiga.

Paki arhiiv enda valitud kohta lahti. Tulemuseks on kataloog nimega `blender-2.56-beta-linux-glibc27-ARCH`, millest leiad `blender`-nimelise käivitava faili.

Blenderi käivitamiseks ava käsuriid, mine Blenderi kataloogi ja kirjuta `./blender`, sealjuures jälgi, et graafiline liides [X](#) oleks samuti käivitatud.

Põhjalikumad juhised

Lae alla fail `blender-2.5x-beta-linux-glibc27-ARCH.tar.bz2` 2.5 beta lehelt (aadressil <http://www.blender.org/download/get-256-beta/#linux386>). Asenda ARCH oma arvuti arhitektuuritähisega — `kasi686` või `x86_64`. Pane tähele, et versiooni number muutub *kindlasti* koos uue Blenderi versiooniga. Vali see, mis kattub sinu süsteemiga. Kui soovid, nimeta kataloog `blender-2.##-beta-linux-glibc#.#-ARCH` millekski lühemaks, nt `blender-2.5`.

Blender on nüüd paigaldatud ja seda saab käsurealt käivitada, kui kirjutad `/blenderi/kataloog/siin/blender` ja vajutad enterit. Kui kasutad KDE-d või Gnome'i, võid Blenderi käivitada failihaldurist, kui lähed kataloogi, kus on fail [Template:Literalblender](#) ja teed sel topetklõpsu.

Võid programmi panna kataloogi `/usr/local` või `/opt` kausta. Kui paigutasid faili oma kodukataloogi (ja nimetasid kausta ümber 'blender-2.5'-ks, käivita kopeerimiseks terminalis käsk:


```
sudo mv ./blender-2.5 /usr/local/
```

Lisaks soovid ilmselt kasutaja PATH-i nimelist keskkonnamuutujat täiendada, et saaksid Blenderit käivitada igast kataloogist.

Kui kasutad aknahaldurit Sawfish, võid lisada rea (`"Blender" (system "blender &")`) oma `.sawfish/rc` faili.

Kuidas KDE all Blenderi ikoon tekitada

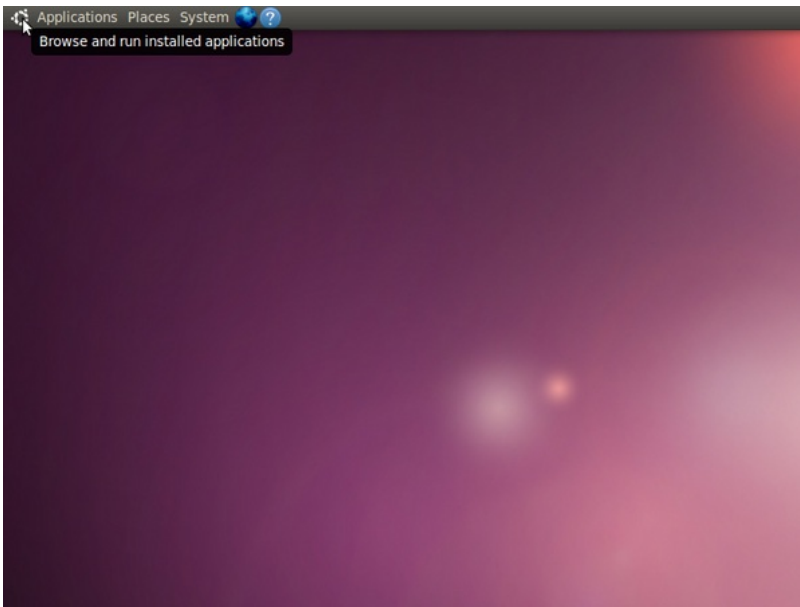
1. Vali System menüüst valik [Template:LiteralMenu Editor](#). See paikneb K menüü all.
2. Vali menüüde nimistust alammenüü Graphics.
3. Klõpsa nupul New Item. Avaneb dialoogiaken, mis küsib nime. Kirjuta sobiv nimi ja klõpsa OK.
4. Satud menüüde nimistusse ja alammenüü Graphics all tõstetakse esile sinu uus sissekanne. Vaata, et paremal pool oleks täidetud järgmised väljad: Name (nimi), Comment (kommentaari), Command (käivituskäsk), Type (tüüp) ja Work Path (töökataloog).
 - Name väli peaks olema juba täidetud, aga saad seda siin muuta.
 - Täida Comment väli. See ütleb, mis abiinfo ilmub, kui hiirega ikooni kohal oled.
 - Klõpsa kausta ikoonil Command välja taga ja otsi Blenderi programmi kataloog. Vali programmi ikoon ja klõpsa OK, mis viib su menüüredaktorisse tagasi.
 - Tüübiks peaks olema Application.
 - work path peaks olema sama mis Command, aga ilma programminimeta lõpus. Näiteks: kui command väljal on `/home/user/blender-2.55-beta-linux-glibc27-ARCH/blender`, peaks Work Path olema `/home/user/blender-2.55-beta-linux-glibc27-ARCH/`. Kui aga panid programmi kataloogi `/usr/local/blender-2.5`, on asjad muidugi seal.
5. Klõpsa Apply ja sulge menüüredaktor.

Blenderi vite KPaneli peale lisamiseks klõpsa paneeli tühjal alal RMB  ja siis vali Add. Klõpsa Button ja siis Graphics ning vali Blender (või mis iganes sa 3. punktis nimeks kirutasid). Alternatiivina võid klõpsida lahti K-menüü alammenüü Configure Panel, siis valida Add, Button, Graphics ja siis Blender.

Blenderi töölaualikooni lisamiseks ava Konqueror (leiad vaikimisi paneelilt või System alammenüüst K-menüüs ja siis leia programmikataloog sealt, kus selle lahti pakkisid. Lohista programmi ikoon Konquerorist töölaual tühjale alale. Sult küsitakse Copy Here, Move Here või Link Here; vali Link Here.

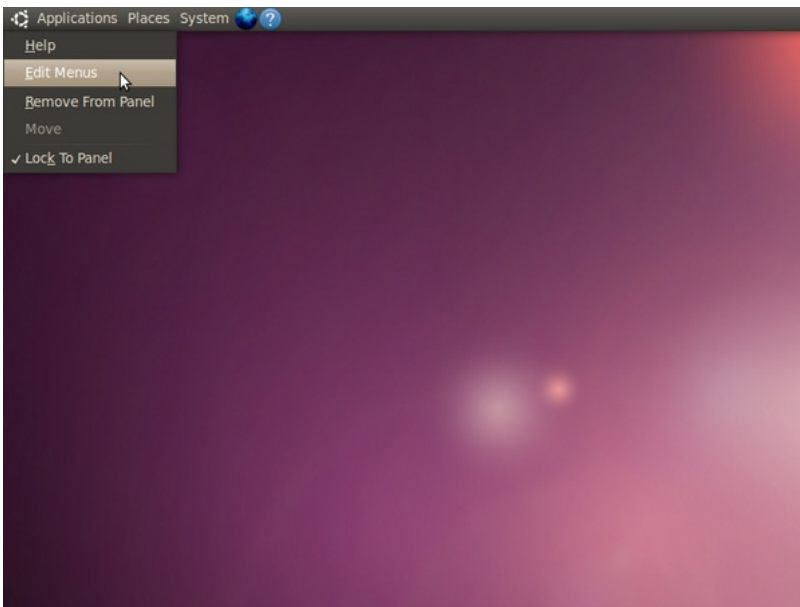
Kuidas Blenderi ikooni GNOME'i all lisada

- Klõpsi RMB  Gnome'i Main Menu paneelil (sõltuvalt teemast ja keelest võib Main Menu veidi teistsugune välja näha)




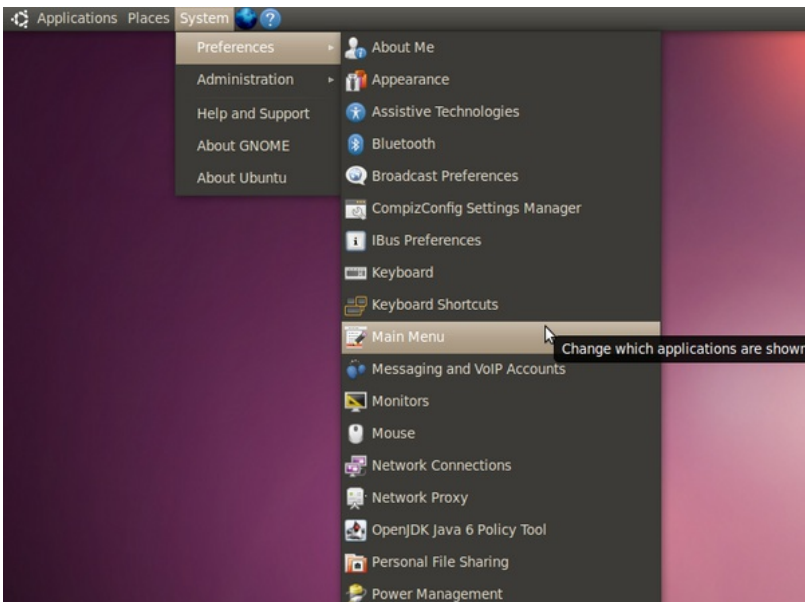
Gnome'i peamenüü paneel

ja vali ilmuvast menüüst Edit Menus,



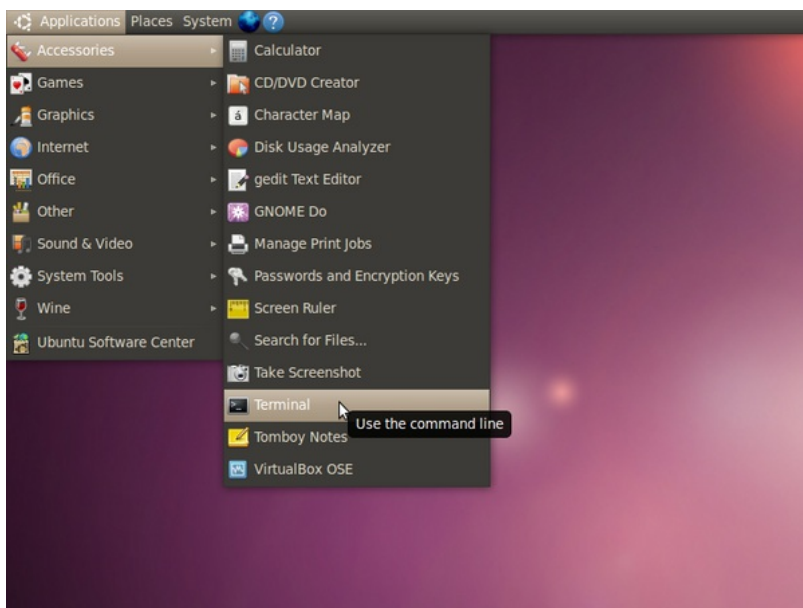
Menüüvalikud paremkliipsul

või tee vasakklõps LMB  Gnome'i paneeli menüül Main Menu ja navigeeri System > Preferences > Look and Feel > Main Menu (Kui su menüü veidi erineb, katseta alltoodud viisi).



Main Menu redaktori asukoht

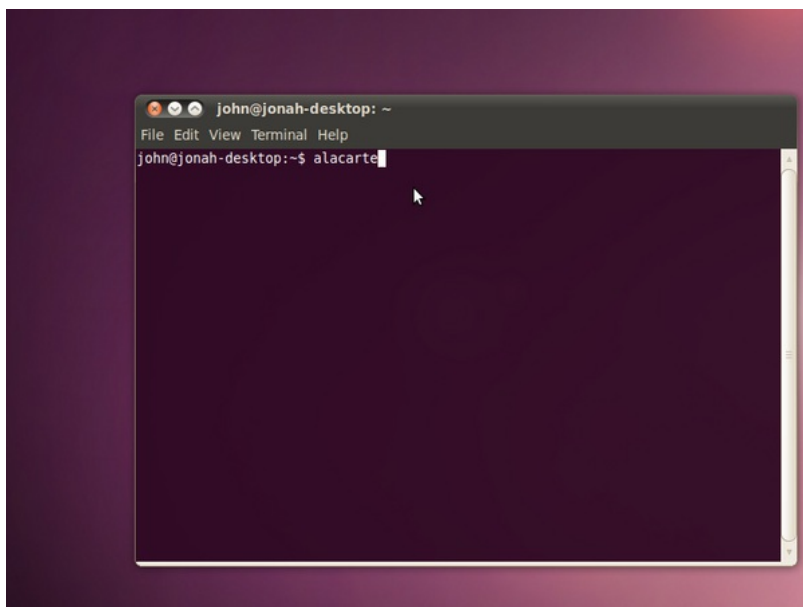
- Teine viis Gnome Main Menu redaktorile ligipääsuks on avada Terminali/Konsooliaken/xtermi aken



Gnome Terminali asukoht

ja kirjutada järgmist:

- `alacarte` <Vajuta ENTER>



Avatud Gnome Terminali aken

- Pärast ühe ülaltoodud meetodi kasutamist oled (loodetavasti) Main Menu redaktoris.



Menüüredaktori aken (alacarte)

- Vali Main Menu dialoogiaknast [Template:LiteralGraphics](#) alammenüü (või see kategooria, kuhu Blenderi ikooni asetada soovid),



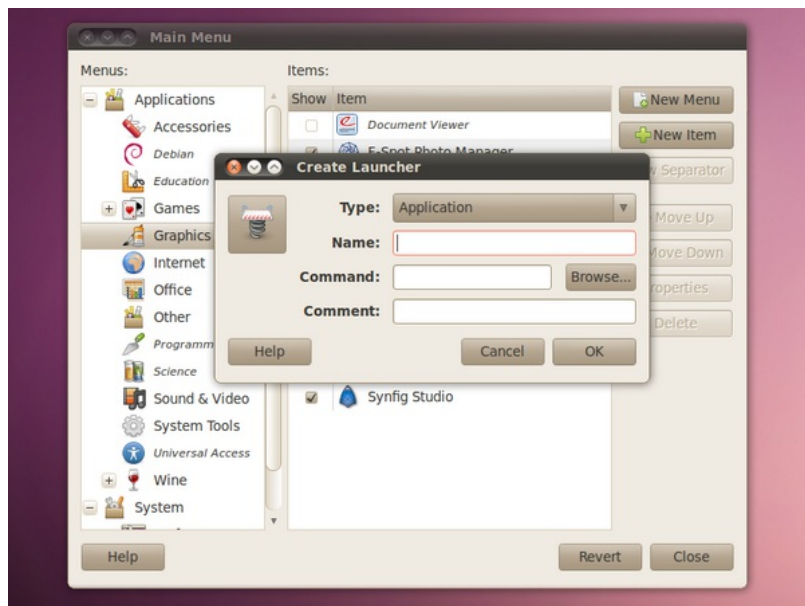
Menüüredaktori aken (alacarte) pärast Graphics valikut

siis klõpsa nuppu New Item.



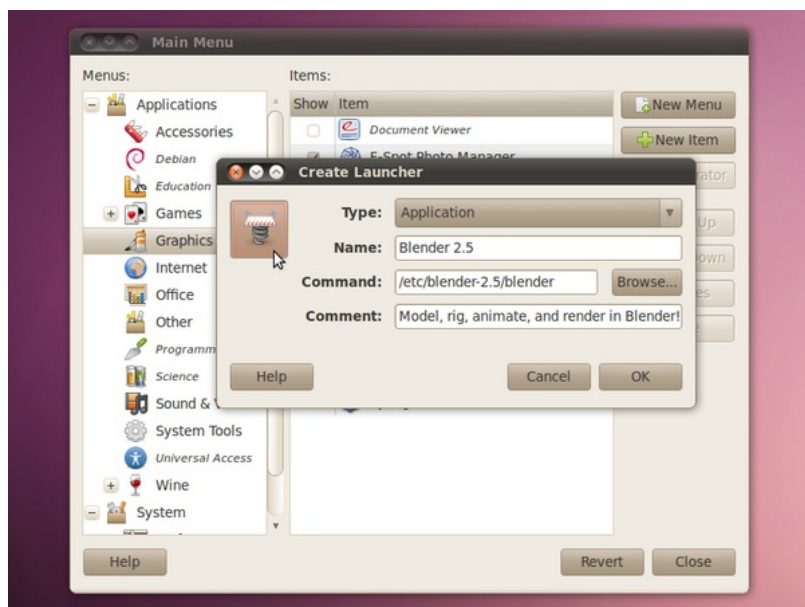
Menüüredaktori aken (alacarte) pärast New Item-nupule vajutamist

- Dialoogiaknas Create Launcher jälg, et Type: all oleks valitud Terminal. See võimaldab näha mõningaid Blenderi veateateid, mis terminali tekivad (seda nimetatakse Blenderi siseselt ka konsooliaknaks).



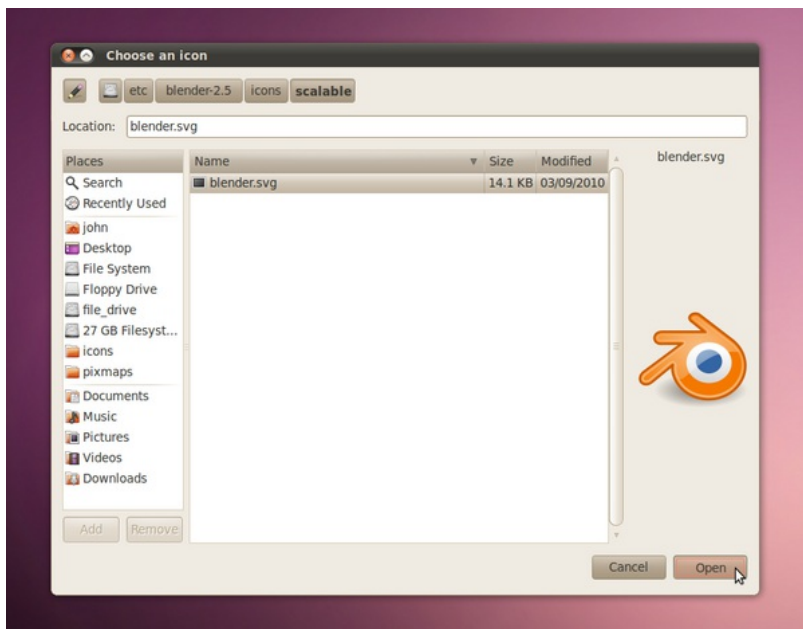
Dialoogiaken Create Launcher (alacarte)

- Create Launcher dialoogiaknas täida järgmisena väljad Name: (nimi), Comment: (kommentaar) ja Command: (käivituskäsk). Kirjuta väljale Name: programminimi, näiteks Blender. Võid selle nimetada kuidas tahad, seda näidatakse menüüs, aga programmi tööd see ei mõjuta. Soovi korral täida Comment: väli kirjeldusega. Seda näitatakse vihjena, kui hiirega menüüsissekande kohal oled. Täida Command: väli käivituva faili blender asukohaga. Näiteks: `/home/user/blender-2.55-beta-linux-glibc27-ARCH/blender` (või hoopis `/usr/local/blender-2.5/blender`, kui programmi sinna lahti pakkisid).
- Klõpsa ikooni nuppu ja vali ikoon (ikooninupp on vaikimisi dialoogiaknas ülal vasakul ja näeb välja nagu vedruga plaat (sõltuvalt süsteemist võib olla ka midagi muud) või kui teemapoolset ikooni pole, on seal kiri No Icon).





Ikoonivalik (alacarte)

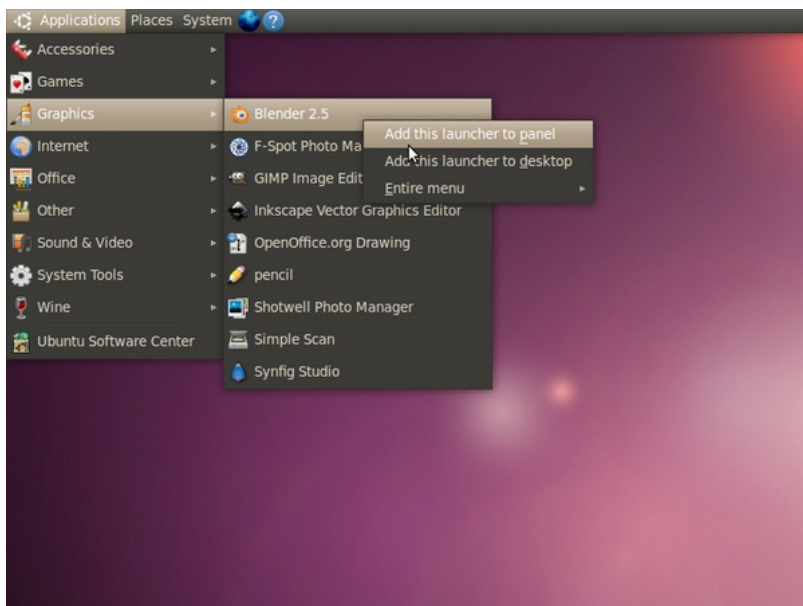
Blenderi ikooni valimiseks klõpsa ikooninuppu ja mine kataloogi, kuhu Blenderi paigaldasid. Sealt mine kausta `icons/scalable/blender.svg`.



Brauseridialoog, mis näitab valitud kataloogis olevaid ikoone (alacarte)



- Kui sa Blenderi ikooni ei leia, võid kasutada 2.45 versiooni ikooni , klõpsa lihtsalt pildil, salvesta see arvutisse ja otsi kettalt üles.
- Siis klõpsa Ok nupul ja menüüredaktori aknasse tekib vastav sissekanne. Kontrolli, et Show item oleks uue sissekande kõrval märgitud.
- Klõpsa Close nupul ja sulge Main Menu redaktor.
- Gnome Menu peaks nüüd laskma ikooni abil Blenderit käivitada.
- Blenderi jaoks Paneeli ikooni lisamiseks klõpsa LMB  Main Menu paneelil ja mine Blenderi menüüsissekande juurde, siis klõpsa RMB  ja vali Add this launcher to panel. Kui see on tehtud, peaks paneelile ilmuma Blenderi ikoon.



Paremklops Blenderi ikoonil (alacarte)

- Töölauaikooni lisamiseks käitu nagu paneelile lisades, aga vali menüüst Add this launcher to panel asemel Add this launcher to desktop.

Kompileerimisjuhised Blender 2.5 jaoks Inteli Mac-ile

Blenderi rakenduse ehitamiseks on kolm sammu. Esiteks tuleb ette valmistada blenderisõbralik keskkond, teiseks kompileerimine ja kolmandaks puhastamine.

Ettevalmistused

Unixi taolises süsteemis kompileerimiseks vajad tööriistadeks programmikomplekti Apple Developer Tools ja MacPortsi süsteemi.

Esmane allalaadimine ja paigaldus

- Lae alla ja paigalda Apple'i [Developer Tools](#)
- Tee tagavarakoopia, nagu juhendab see (inglisekeelne) lehekülg <http://www.mac-how.net/> Backup your Hard Drive with Time
- Laadi ametlikult kodulehelt [MacPorts](#)

Järgmisena vajad subversionit ja sconsi. Kirjuta:

```
$ login
# sudo port selfupdate
# sudo port install subversion scons
# logout
```

Login annab administraatoriõigused ja vajab administraatori parooli. Pärast vajalike eelduste kompileerimist ja pakettide paigaldust võid Blenderi läheteteksti alla laadida.

Blenderi kompileerimine

Eeldame, et eelnevad sammud on tehtud ja kõik on kulgenud edukalt. Muidu järgnev ei tööta!

Lae lähtetekst ja teegid

Kirjuta:

```
$ export PATH=/Developer/usr/bin/:$PATH
$ cd ~
$ mkdir ~/blender-build
$ cd blender-build
$ svn co https://svn.blender.org/svnroot/bf-blender/trunk/blender blender
$ svn co https://svn.blender.org/ \
  svnroot/bf-blender/trunk/lib/darwin-9.x.universal lib
```

Kohanda seadistuste fail

Lähtetekstid on nüüd ~/blender ja ~/lib all. Ava oma lemmikredaktoris (TextEdit.app või vi ...) fail darwin-config.py:

```
[fuzzy]$ cd ~
$ vi ~/blender/config/darwin-config.py
```

Muuda MACOSX_ARCHITECTURE x86_64 peale:

```
MACOSX_ARCHITECTURE = 'i386' -> MACOSX_ARCHITECTURE = 'x86_64'
```

Lisa allapoole paar rida:

```
else :
    MAC_MIN_VERS = '10.5'
    MACOSX_DEPLOYMENT_TARGET = '10.5'
    MACOSX_SDK='/Developer/SDKs/MacOSX10.5.sdk'
    #LCGDIR = '../lib/'
    CC = 'gcc-4.2'
    CXX = 'g++-4.2'

## add this
BF_NUMJOBS = '2'
pwd = 'pwd'
pwd_res=commands.getoutput(pwd)
LCGDIR = pwd_res + '/../libs'
##
```

Blender.app ehitamine

Pärast faili salvestamist mine kausta ~/blender ja kirjuta:

```
$ svn update
$ scons -h
$ scons
```

Ehitamine on nüüd alanud ja peaks edukalt lõppema. Kui nii juhtub, leiad blender.app faili siit ~/blender/install/darwin/blender.app.

Puhastamine

Pärast faili blender.app oma tavalisse rakenduste kausta salvestamist võid süsteemi puhastada. Kustuta kõik kasutusel olnud failid ja kataloogid. Selleks jooksuta käsud:

```
$ mv ~/blender-build/install/darwin/* ~/Blender-2.5/
$ cd ~
```

```
$ rm -rf ./blender-build
$ login
# sudo port -f uninstall installed
# sudo rm -rf \
/opt/local \
/Applications/DarwinPorts \
/Applications/MacPorts \
/Library/LaunchDaemons/org.macports.* \
/Library/Receipts/DarwinPorts*.pkg \
/Library/Receipts/MacPorts*.pkg \
/Library/StartupItems/DarwinPortsStartup \
/Library/Tcl/darwinports1.0 \
/Library/Tcl/macports1.0 \
~/macports
```

Teised toetatud operatsioonisüsteemid

FreeBSD

Laadi fail `blender-2.##-FreeBSD-####.tbz`, mille leiad [Blenderi veebi](#) alajaotusest "Downloads". 2.## on Blenderi versioon ja #### on masina arhitektuur (i386 või amd64).

Paki arhiiv lahti. Blenderi käivitamiseks mine käsureale ja kirjuta `./blender`, sealjuures jälgi, et graafiline liides [X](#) oleks samuti käivitatud.

Teised ametlikult mittetoetatud operatsioonisüsteemid

MorphOS

Selle platvormi porti haldab Guillaume Roguez. Lisainfot leiab [arendaja wikist](#) ([prantsuse versioon](#)).

Selle pordi paigaldamine eeldab, et pakid arhiivi lahti kõvakettale ja mitte RAM-i, sest Python ei suuda oma dünaamilisi .pym mooduleid RAM-ist kasutada.

Blenderi seadistamine

Blenderi vaikimisi paigaldus pakub palju funktsioone ja näeb üsna lahe välja. Kui programmi paigaldad või täiendad, on paar asja, mida kõigepealt teha tasub:

- Näita Blenderile kätte, kus oma masinas asju hoiad.
- Kopeeri lisandunud Pythoni skriptid ja kontrolli, kas nad ikka veel töötavad.
- Ütle Blenderile, kus on videotöötlus- ja tekstuuripuginad.
- Kohanda oma animeerimise, modelleerimise, materjalide, videoribade järjestaja (*Sequence Editor*) ja skriptimise aknad.
- Määra animatsioonide salvestamiseks vaikimisi kataloog.

Kõige ülemine aken sisaldab kõiki [seadistusi](#), muuhulgas sakki File Paths, mida kohendada tuleks. Seejärel mine kataloogi `/Blender/Scripts/` ja kopeeri kõik vajalikud skriptid ümber kodukataloogis olevasse kausta `.blender/scripts`. Tekstuuri ja järjestaja pluginate kataloogiseadistused on kaardil User Preferences, soovitatav on neid hoida `/Blender/bin` all igaüks omas kataloogis. Oma erinevad töökeskkonna paigutused leiad ekraani ülaservas paiknevast rippmenüüst. Igaüht neist on võimalik oma äranägemise järgi suurendada või ümber seadistada (vaikimisi seadistus sobib algajatele hästi). Kui klõpsid ülaservas nupul Render, salvestatakse animatsioonid seadistustes määratud ülemisse Render kausta ja võib juhtuda, et soovid seda oma ajutiste failide kausta (`temp`) suunata. Kui oled valmis, saad muutused salvestada CtrlU abil.



Vihje

Nupukombinatsioon CtrlU salvestab kõik parajasti avatud Blenderi faili seadistused vaikimisi avatavasse Blenderi faili (mille nimi on harilikult `.B.blend`). Neid vaikimisi seadeid loetakse siis, kui Blender käivitub või kui tühja faili saamiseks vajutatakse CtrlX. Kui kogemata oma vaikeseadete failis olevad sätted ära muudad, on algeise taastamiseks järgmised viisid:

- Vali menüüst File valik Load Factory Settings ja kui see on tehtud, vajuta CtrlU, mis salvestab algsed seadistused Blenderi vaikefaili. Kui sul on Blenderi vanem versioon, ei pruugi see meetod töötada, sellisel puhul proovi järgmist meetodit.
- Kustuta `.B.blend` fail (asukoht sõltub operatsioonisüsteemist); kui Blender taaskäivitada, luuakse see vaike seadistusi kasutades uuesti.

Enda kataloogistruktuuri määramine

Kui oled Blenderi seadistamisel algaja, peaksid oma töö kohe algul organiseerima, sest sul tekib uusi tekstuure, mudeleid, pilte, `.blend` faile, `.zip` faile, [skripte](#) jne. Kõigi nende ühte kataloogi kokkutopppimine tekitab segadust ja seega oleks tark võtta hetk aega ja selle kõige organiseerimiseks paar kataloogi teha. Järgnev on aastatepikkuse kogemuse kulminatsioon, aga see ei ole andmete organiseerimiseks kindlasti ainus viis. Suuremate projektide haldamiseks on tasuta tööriistu (näiteks [CVS/Subversion](#) ja [Verse](#)), nagu ka failihaldussüsteeme, aga need on väljaspool selle käsiraamatu temaatikat.

Tavakasutajale on soovitatav kataloogistruktuur järgmine:

- `C:\Blender` – Jagatud kataloog, milles on järgmised alamkaustad:
 - `\bin\` – Kõik binaarsed failid (paigaldamise `.exe` failid), utiliidid ja lisad, nagu näiteks [YafRay](#), [Python](#), Gocubic, [Panocube](#), [Virtual Dub](#) jne.
 - `\examples\` – Teiste poolt tehtud tööd (pildid ja filmid), mida soovid võrgust eemal olles uurida.
 - `\lib\` – Viitematerjalide kogu (sellest täpsemalt hiljem).
 - `\man\` – [Käsiraamatud](#), [pdf-juhendid](#) nagu Blender Basics, videod ekspertidelt, [spikrikaardid](#) ja enda kirjutatud märkmed keerulisemate asjade hiljem meenutamiseks.
 - `\play\` – Mänguväljak nende `.blend` failide salvestamiseks, millega niisama mässad.
 - `\script\` – Need [Pythoni skriptid](#), mis Blenderiga kaasa ei tule.
 - `\tmp\` – Ajutised failid ja ajutine väljund.
 - `\tut\` – Veebist korjatud õppematerjalid. Veebis on palju video- ja veebimaterjali (salvesta täieliku veebilehena).
 - `\util\` – Blenderi utiliidid nagu [Make Human](#), [World Forge](#) või [Tree Generator](#).
 - `\work\` – Ja lõpuks kõige olulisem: kui tõesti mõnest tähendusrikkamast projektist peaksid kinni haarama, mis võib-olla mänguväljakult välja kasvas, saad ta siia panna.



Vihje

Kaustade nimed on kasulik luua inglise keeles, sest nii saad harjumuspäraseid katalooginimesid kasutada ka rahvusvahelistes projektides, ilma et peaksid ümber harjuma. Samuti mõistavad sulle appi tulnud välismaalastest sõbrad su kataloogistruktuuri.

Selgitused kataloogidele

Põhikaust on `/Blender/`, mida Windowsi all võib `/Shared Downloads/` kaustas hoida. Selle alla loo alamkataloog `/Blender/bin/`, mis hoiab käivituvaid faile või `.exe` paigaldusfaile, nagu ka teisi Blenderi poolt kasutatavaid programme, näiteks [YafRay](#)d, ja mõningaid mugavad DLL-e, mida Blenderi täiendamisel kasutada võid.

- **Library (andmekogu):** On ilme, et soovid ise maailma luua, aga tegelikult on võrgus päris palju mudeleid ja kraami, mida muu loomerahvas juba valmis on teinud. Kogu selle valmishetatud info hoidmiseks on hea teha andmekogu kataloogi `/Blender/lib/`. Selle all võiks olla alamkaustad `/mesh` (võrede (*mesh*) hoidmiseks), `/tex` tekstuuripiltidele ja `/pic` piltidele, näiteks neile, mida kasutad eeskujudena. Mudelite `/blender/lib/mesh` kaustas on palju alamkaustu, näiteks `/animal`, `/human`, `/machine` ja `/house` (vastavalt loom, inimene, masin, maja), mis kõik hoiavad vastavate asjade näiteid ja mudeleid. Kataloog `/tex` sisaldab sarnast kaustade komplekti ja hoiab `.jpg` faile ja `.blend` faile, milles on objektide kaunistamiseks ja värvimiseks mõeldud materjalid. Kataloog `/tex` all on alamkataloogid `/nature`, `/buildings`, `/painted` ja `/metal` (vastavalt loodus, ehitised, värvipinnad, metall). Kataloog `/pic` sisaldab näidismaterjali inimestest (Angelina Jolie), nägudest (su tütar), mõõblit, su autot (Dodge Viper) ja muud pildimaterjali ja kontseptuaalkunsti modelleerimisel eeskujuks võtmiseks.
- **Käsiraamat ja kasutusjuhendid:** Tee kaust `/Blender/man/` ja pane sinna käsiraamatud ja juhised, olgu nad siis `.html`, Wordi (`.doc`) või `.pdf`-formaadis. Neid on leida üsna suurtes kogustes. Lisaks saad sinna kausta panna kohaliku koopia nendest samadest wikilehtedest, juhiks kui töötad internetita.
- **Õppematerjalid:** Õppematerjale on allalaadimiseks palju. Loo kaust `/Blender/tut/` ja pane head õppematerjalid sinna hoiule.

Mõned õppematerjalid on inimeste endi üles pandud ja kaovad kergesti, seega kui leiad artikli, mis sind aitab, säilita seda siin kaustas.

- [Pythoni skriptid](#): Blenderi funktsionaalsuse laiendamiseks kasutame skriptimiskeelt [Python](#). Selles keeles [skripte](#) on kümneid. Kui leiad skripti, salvesta see kausta `/Blender/script/`. Sinna pane ka kõik tagavarakopeerimiseks kasutatavad käsujadad jne.
- **Utiliidid**: Blender on arenenud nii kaugele, et on olemas terved imelisi asju loovad programmid. Pane oma [Make Human](#) ja [World Forge](#) kataloogi `/Blender/util/`.
- **Teeme ära!**: Nüüd on *sinul*, mu noor õpilane, oma isiklikku ruumi tarvis. Loo kaustad `/Blender/play/` ja `/Blender/work/`, et sul oleks koht, kus lustida ja mängida, ning koht, kuhu saad panna oma tööfailid. Olen Blenderis teinud reklaami, dokumentaalfilmi Nigerist ja ühe patendi ([#6.796.205](#)) ja iga projekti jaoks on `/Blender/worki` all oma kataloog. Nende sees on komplekt `/tex`, `/pic`, `/render`, ja `/wav` kaustu, mis hoiavad vastavalt tekstuure, pilte, renderdamise tulemusi ja helifaile. Päris `.blend`-failid on kataloogis `/work/xxx`, kus `xxx` on projekti lühike nimi. Kataloog `/Blender/play/` on üsna organiseerimata ja seal on [Yafray](#), `anim` (animatsioon), `Lightning` (valgustamine) ja muud kaustad – sisuliselt on tegemist prügihunnikuga, kus tuleb siis sorida, kui on tarvis teha midagi, mis kunagi juba on tehtud, aga pole meeles, kuidas see täpselt käis.

Internetiühendus

Blender on veebiga integreeritud ja tal on kaks faili, mis võivad soovida võrku külastada ja sellega tulemüüri aktiveerida:

- `Blender Foundation\Blender\blender.exe` – Tuumprogramm.
- `Blender Foundation\Python\pythonw.exe` – Pythoni skriptide interpretaator.

Lisaks võib Blender käivitada veebibrauseri (näiteks Firefox), kui soovid näiteks Help » Manual abil käsiraamatut sirvida.

Internetiühendus pole Blenderi igapäevakasutuseks tarvilik. Samas on käsiraamatu ja muude võrguressurssidega tutvumiseks internetiühendust mõnikord vaja.

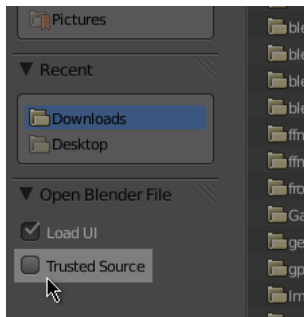
Turvalisus

Blender on mõeldud stuudiokasutuseks ja sõltub Pythonist, mis on võimas skriptimiskeel. Pythoniga on võimalik luua Blenderile uusi tööriistu, importijaid ja eksportijaid; lisaks kasutada animatsiooni toetamiseks. Blenderit on skriptidega võimalik lõputult täiendada.

Osa Pythoni võimsusest tuleb sellest, et tal on su süsteemile täielik ligipääs, paraku on seda võimu võimalik ka kurjalt ära kasutada. On võimalik (kuigi selgelt tavatu), et ebaausad inimesed jagavad .blend-faile, milles olevad skriptid võivad su süsteemi kahjustada. Skripte saab kinnitada failide külge sellisel moel, et nad lähevad käima siis, kui .blend-fail avatakse.

Ettevaatust kahtlastest allikatest .blend-faile või tööriistu laadides!

Kaitse



Kuritahtlike .blend-failide eest on end võimalik kaitsta nii, et keelad skriptide käivitamise .blend-failide avamisel. See tähendab, et Pythonit kasutavad tööriistad või diginukud ei tööta, kuid see pole probleem, kui kasutad .blend-faile, mis sellist funktsionaalsust ei vaja (näiteks materjalide kogud); samuti on sul võimalik nii faile põhjalikumalt uurida ja failis olevaid riske hinnata.

Vaikimisi usaldab Blender kõiki faile ja laseb skriptid automaatselt käima. Kui sa faili ei usalda ja tahad lisakaitset, võid keelata *Trusted source* valiku File->Open dialoogiakna alumises vasakus nurgas. Kui faili ei usaldata, siis failis olevaid Pythoni skripte ei käivitata.

Blenderi esmakordne käivitamine

Kui sa juba tunnud Blenderi versiooni 2.4 või mõnda muud 3D-programmi nagu Maya, 3ds Max või XSI, märkad sa kohe, et Blender 2.5 on harjumuspärasest märksa erinev. Siiski näed sa varsti ka oma eelmiste programmidega sarnaseid elemente, näiteks 3D-vaade, struktuuriredaktor ja ajatelt. Kui see on sulle esimene kord mõnda 3D-programmi kasutada, võid tunda end pisut segaduses. Õnneks on Blenderis 3D-modelleerimise õppimisel tegelikult ainult üks reegel: ära karda uurida ning katsetada!

Pärast Blenderi käivitamist heida pilk avaekraanile, kus sa näed paremas ülemises nurgas Blenderi versiooninumbrit.



Vasakul küljel on näha mõned kasulikud lingid, nagu sinu kasutatava versiooni [väljaandelogi](#) (mis on selles versioonis uut), [Wiki käsiraamat](#) (see, mida sa praegu loed) ja [ametlik Blenderi koduleht](#). Need lingid saab kätte ka menüüst Help (abi).

kiirklahvide variante (vaikimisi on võimalik valida Blenderi ja Maya vahel).

Paremal küljel on nimekiri viimastest Blenderi (.blend) failidest, mis sa oled salvestanud. Kui sa käivitasid Blenderi esimest korda, on see tühi. Selle nimekirja saab kätte ka menüüst File » Open Recent (failid->ava viimatised). Menüüst "interaction" (suhtlus) saad sa valida eelhäälestatud

Et alustada tööd Blenderis, on sul kolm võimalust:

- Klõpsa mõne hiljutise faili nimele (kui sul neid on)
- Klõpsa kuhugi ekraanil (välja arvatud avaekraani tume ala) või
- Vajuta Esc ja alusta uue stseeniga

Salvesta oma töö regulaarselt

Blender ei hoiata programmi sulgemisel, kui sul on salvestamata muudatusi, seega salvesta sagedasti! Kui sa siiski sulged Blenderi ilma viimaseid muudatusi salvestamata, ei ole kõik veel kadunud. Lihtsalt ava Blender uuesti ja klõpsa avaekraani nupul Recover Last Session (taasta viimane sessioon). Sa leiad selle valiku ka menüüst File » Recover Last Session (fail->taasta viimane sessioon).

Ajutine .blend-fail

Iga kord, kui Blender töö lõpetab, salvestab ta käsiloleva dokumendi ajutisse .blend-faili. Kui sa viimase sessiooni taastad, loeb Blender andmed sellest failist.

Kasutajaliidese põhialused



Blender on multiplatvormprogramm: see tähendab, et ta töötab Linuxi, Mac OS X-i ja Windowsi all. Et Blenderi kasutajaliides baseerub [OpenGL](#)-i graafikateegil, võid sa märgata, et kasutajaliides näeb kõigis põhilistes operatsioonisüsteemides ühesugune

välja.

3 reeglit

Blenderi kasutajaliides tugineb kolmele põhiprintsiibile:

- **Mittekattuvus:** Kasutajaliides laseb sul ühe pilguga näha kõiki olulisi valikuid ja tööriistu ilma aknaid edasi-tagasi lohistamata⁽¹⁾.
- **Mittetökestamine:** Tööriistad ja kasutajaliidese elemendid ei tõkesta sul teiste Blenderi osade kasutamist. Blender ei viska ekraanile dialoogiaknaid, mis nõuavad kasutajalt andmete sisestamist enne mingi tegevuse algust.
- **Kontekstivaba:** Kasutajaliides peab olema nii püsiv ja etteaimatav kui võimalik ja mitte muutma lennult oluliste sisestusmeetodite (hiir, klaviatuur) tähendusi.

⁽¹⁾Siiski lubab Blender 2.5 avada korraga mitu akent tööks mitme monitoriga. See on erand *mittekattuvuse reeglile*.

Võimas kasutajaliides



Blenderi kasutajaliides joonistatakse täielikult [OpenGL](#)-i abil. See võimaldab sul teda vastavalt oma vajadustele ümber kohandada. Aknaid ja teisi elemente saab nihutada, suurendada ja nende sisu ümber paigutada. Sa saad teha sobiva ekraanipaigutuse iga erineva tööprotsessi jaoks, anda sellele nime ja salvestada.

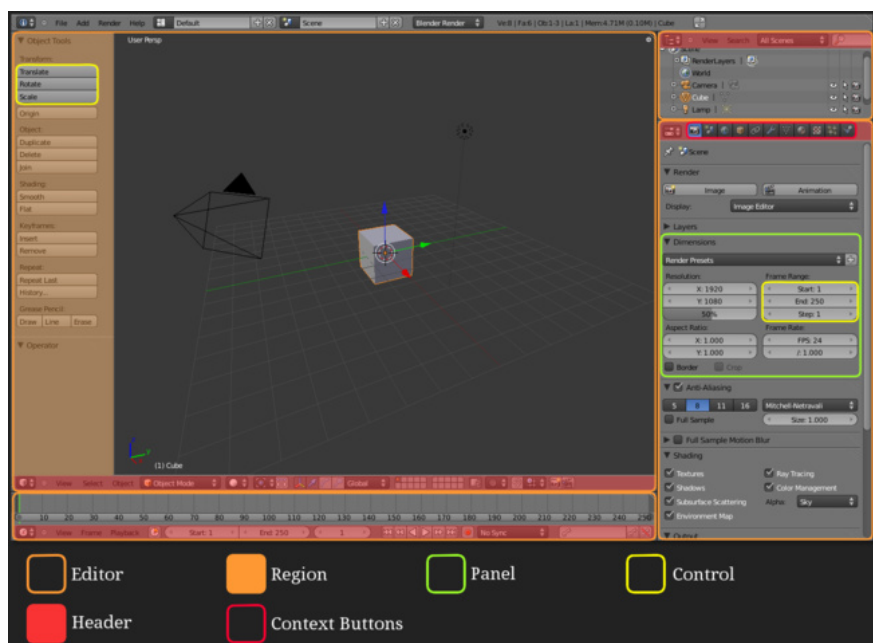
Et sinu tööd kiirendada, kasutab Blender laialdaselt kiirklahve. Et hõlbustada meeldejätmist, saab kiirklahvide tabelit ise muuta.

Ülevaade

Heidame pilgu kasutajaliidese vaikimisi valitud ülesehitusele. See koosneb redaktoritest, päistest, kontekstinuppudest, paneelidest ja seadistuväljadest.

- Blenderis nimetame me **redaktoriks** (Editor) spetsiifilise funktsiooniga programmi osi (3D-vaade (3D View), omaduste redaktor (Properties Editor), videomonteerija (Video Sequence Editor), sõlmeredaktor (Nodes Editor)...). Igal redaktoril on üleval või all eraldi *päis*.
- **Kontekstinupud** (Context buttons) võimaldavad ligipääsu seadistustele. Nad on nagu järjekohajad ja on tihti paigutatud redaktori päisesse (näiteks omaduste redaktoril (Properties Editor)).
- Et kasutajaliidest loogiliselt organiseerida, on igal neist redaktoritest valikud paigutatud **paneelidele** (varjupaneel (Shadow), värvipaneel (Color), mõõtmetepaneel (Dimensions)...).
- Osadel redaktoritel on veel **külgribad** (Sidebars). Sel juhul on paneelid ja seaded paigutatud nendele. Töölaua paremaks organiseerimiseks on külgribasid võimalik ajutiselt peita.
- Paneelidel asuvad **seadeväljad** (Controls). Need võimaldavad sul muuta kasutatavaid meetodeid, teha valikuid ja sisestada väärtusi. Blenderis on hulk seadeväljade tüüpe:
 - **Nupud:** Annavad juurdepääsu mõnele tööriistale (nihutamine (Translate), pööramine (Rotate), võtmekaadri lisamine (Insert Keyframe)). Töö kiirendamiseks on need tööriistad enamasti kättesaadavad ka kiirklahvidega. Kiirklahvi nägemiseks liiguta hiir nupu kohale ja Blender näitab lühivihjet selle kohta.
 - **Linnukesed (Checkboxes):** Võimaldavad valikuid sisse ja välja lülitada. Vastavatel seadetel on ainult kahendväärtused (jah/ei, 1/0).
 - **Numbriväljad (Sliders):** Võimaldavad sisestada ujukomaarve. Need võivad olla piiratud (näiteks 0.0 kuni 100.0) või mitte ($-\infty$ kuni $+\infty$). Pane tähele, et Blenderis on kahte tüüpi numbrivälju.
 - **Valikumenüüd:** Võimaldavad valida väärtuse etteantud nimekirjast. Valikumenüüde ja linnukeste erinevus on see, et siin on väärtustel nimed ja neid saab olla rohkem kui kaks.

[Loe lähemalt nuppude ja seadete kohta »](#)



Sisendi seadistamine

Blenderi kasutajaliides on mõeldud kasutamiseks järgneva soovitusliku sisendseadmete komplektiga:

- Kolme nupuga kerimisnupuga hiir
- Täisklaviatuur eraldi numbriväljaga
- NumLock peaks tavaliselt olema välja lülitatud.




Kui sul ei ole soovituslikke sisendseadmeid (näiteks kui kasutad sülearvutit), saad lülitada Blenderi kasutajaeelistustes sisse nende emuleerimise olemasolevate seadmete abil.

[Loe lähemalt Blenderi seadistamisest »](#)



Selles käsiraamatus kasutatavad tähistused

Siin käsiraamatus on kasutaja poolt antavate sisendkäskude jaoks järgmised tähistused:

- Hiirenuppude tähised on:

LMB  - vasak hiirenupp (*Left Mouse Button*)
MMB  - keskmine hiirenupp ja (*Middle Mouse Button*)
RMB  - parem hiirenupp (*Right Mouse Button*)

- Kui sinu hiirel on kerimisratas

MMB  - märgib rattale vajutamist (nagu nupule), samas kui
Wheel  - märgib rattaga kerimist.

- Kiirklahve näidatakse siin käsiraamatus nii, nagu nad paistavad klaviatuuril. Näiteks:

G - märgib väiketähte "g".

⇧ Shift, Ctrl ja Alt tähistavad üldiselt modifitseerivaid klahve

CtrlW või ⇧ ShiftAltA - märgivad, et need klahvid tuleb korraga alla vajutada

0 NumPad kuni 9 NumPad, + NumPad - märgivad klahve eraldi numbriklahvide väljal. (*Pad* = numbriväli)

Teisi klahve tähistatakse nende nimedega nagu Esc, ⇐ Tab, F1 kuni F12. Pane tähele ka nooleklahve ←, → ja nii edasi.

Üldine kasutus

Blenderis on hiirt ja klaviatuuri vaja pidevalt kõigis tööprotsessides ja seetõttu on Blenderi kasutajate hulgas saanud tuntuks kuldreegel: **Hoia üks käsi hiirel ja teine klaviatuuril**. Kõige sagedamini kasutatavad klahvid on paigutatud nii, et neid saab ladina tähestikuga klaviatuuril vajutada vasaku käe lähteasendist (nimetissõrm klahvil F). See eeldab, et kasutad hiirt parema käega.










Kui kasutad enamasti standardsest ladina paigutusest oluliselt erinevat klaviatuuri, tasub sul võib-olla Blenderis töötamise ajaks määrata klaviatuuri paigutuseks US/inglise. Pane tähele, et sa saad Blenderi vaikimisi kiirklahve ise muuta. Selles käsiraamatus kasutatakse siiski kiirklahvide vaikeväärtusi.



[Loe lähemalt Blenderi seadistamisest »](#)

Hiirenupu emuleerimine

Kui sul ei ole 3 nupuga hiirt, pead sa [Kasutajaeelistustes \(User Preferences\)](#) lülitama sisse kolmanda nupu emuleerimise (vaikimisi välja lülitatud).

Järgnevas tabelis on näidatud emuleerimiseks kasutatavad kombinatsioonid:

3 nupuga hiir		2 nupuga hiir	Apple'i hiir
LMB 	LMB 	LMB  (hiirenupp)	
MMB 	Alt LMB 	⇧ Opt LMB  (Option/Alt-klahv + hiirenupp)	
RMB 	RMB 	⇧ Cmd LMB  (Command/Apple'i klahv + hiirenupp)	

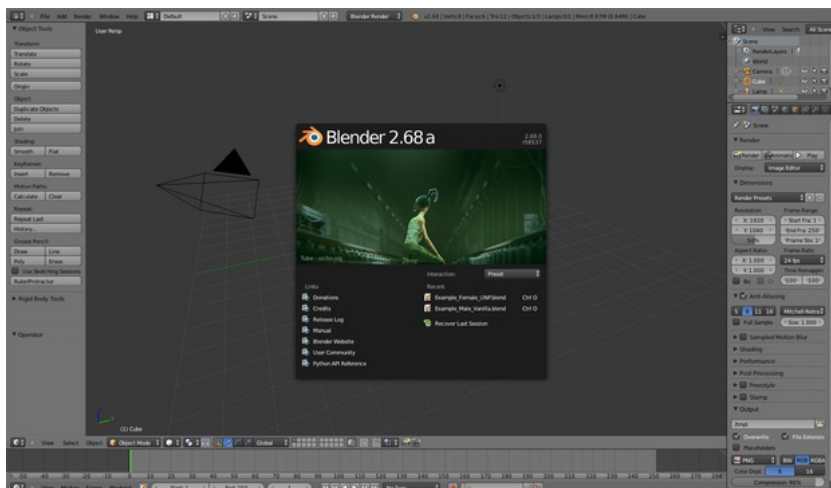
Kõiki selles käsiraamatus toodud hiire- ja klahvikombinatsioone on võimalik sisestada ka siin tabelis toodud kombinatsioonidega. Näiteks tuleb ⇧ ShiftAlt RMB  ühenupulise hiire korral sisestada kui ⇧ ShiftAlt⇧ Cmd LMB .

Numbriklahvide emulatsioon

[Loe numbriklahvide emulatsioonist lähemalt kasutajaeelistuste lehelt »](#)

Aknad

Kui Blenderi käivitad, peaksid sa nägema järgmist pilti (keskmine tervitusaken võib uuemates versioonides olla teistsugune):

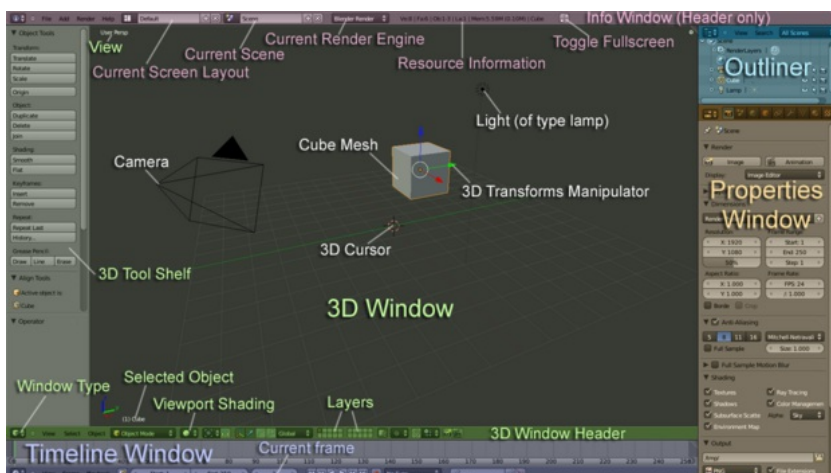


Ekraani keskel on tervitusaken. Siit saab kiiresti ja lihtsalt kätte hiljuti avatud Blenderi failid. Kui tahad alustada tööd uue failiga, klõpsa lihtsalt hiirega väljaspool tervitusakent. Akna kaob ja nähtavale tuleb ekraani vaikepaigutus kuubiga stseeni keskel.

Kõiki nähtavaid aknaid saab jagada iseseisvateks väiksemateks tööaladeks (nagu kirjeldatakse [paanide paigutamise](#) peatükis). Allpool kirjeldatakse vaikestseeni.

Vaikestseen

Vaikestseen avatakse iga kord, kui käivitad Blenderi või lood uue faili.



Blenderi vaikestseen

See koosneb viiest aknast:

- Peamenüü (infoaken) (Info Window) ekraani ülasosas, millest on näha ainult päis (seal polegi midagi muud).
- Suur 3D-aken (3D View)
- Ajatelg (Timeline) ekraani allääres.
- Ülevaateaken (Outliner) paremal üleval.
- Omaduste (Properties) aken (nuppude aken) paremal eelmise all.

Sissejuhatuses selgitame mõnd põhilist elementi.

Infoaken (Info Window) (või peamenüü)


Vaata selgituseks eespool olevat pilti.

- **Parasjagu kehtiv ekraanipaigutus (Current Screen) (vaikevalik on Default):** Blenderil on vaikimisi kaasas mitmed eelseadistatud ekraanipaigutused (Screens), mille vahel valida saad. Kui vajad [kohandatud ekraanipaigutust](#), saad sa selle ise luua ja talle nime anda.
- **Hetkel avatud stseen (Current Scene):** Blender võimaldab kasutada paralleelselt mitut [stseeni](#), st töötada korraga mitmete virtuaalsete maailmadega, mis võivad olla kas täiesti iseseisvad või sisaldada objekte ja/või võresid, mis on mõne teise stseeni objektide lingitud duplikaadid. (Osades 3D-programmides on iga stseen omaette failis, Blender võimaldab aga paigutada ühte .blend-faili mitu stseeni).
- **Valitud renderdamismootor (Current Rendering Engine):** Nimekirja võimalikest renderdamismootoritest.
- **Ressursside kasutus (Resource Information):** Annab ülevaate, kui palju on hetkel kasutuses Blenderi ja süsteemi ressursse. See väli annab teada kui palju mälu stseen kasutab, kui palju on temas tippu, külgi ja objekte ning kui palju elemente on hetkel valitud. See võib aidata tuvastada, kui stseen muutub arvuti riistvara jaoks liiga keeruliseks.

- **Täisekraan (*Toggle Fullscreen*):** Lülitab täisekraani režiimi sisse/välja.

3D-aken (*3D Window*)

Vaata selgituseks eespool olevat pilti.

- **Akna tüüp:** Võimaldab vahetada akna tüüpi (see valik on iga akna päises). Näiteks kui tahad näha selles aknas stseeni struktuuri ülevaadet (Outliner), klõpsa sellele väljale ja vali vastav aknatüüp. Vaata [aknatüüpide täielikku nimekirja](#)
- **3D-teisendaja (*3D Transform Manipulator*):** See on visuaalne abivahend objektide teisendamiseks (lohistamiseks/liigutamiseks, pööramiseks ja skaleerimiseks). Objekte saab teisendada ka kiirklahvide abil: (G/R/S); CtrlSpace vahetab teisendaja nähtavust. Seda viimast saab teha ka vajutades tööriistaribal koordinaatsüsteemi ikoonile. Nihke, pöörde ja mõõtkava teisendajaid saab teha nähtavaks, vajutades vastavalt ühele kolmest teisendaja ikoonist koordinaatsüsteemi ikoonist paremal. ⇧ Shift LMB -klõps ikoonil vahetab vastava teisendaja nähtavust.



- **3D-kursor (*3D Cursor*):** Sellel on mitmeid rakendusi. Näiteks määrab see, kuhu stseenile lisatud uued objektid tekivad, või siis võib ta määrata keskme, mille ümber objekti pööratakse. Siin on näha 3D-kursor ilma ülejäänud stseenita:



- **Kuubi võre (*Cube Mesh*):** Vaikimisi sisaldab uus Blenderi stseen alati kuubikujulist võreobjekti (Mesh), mis asub globaalse 3D-koordinaadistiku keskel (üalloodud pildil on seda liigutatud). Mõne aja pärast tahad ilmselt Blenderi vaikeseadistust muuta. Selleks tuleb [Blender seadistada](#) selliseks, nagu sa tahad, ja siis salvestada see "vaikimisi" konfiguratsioonina, kasutades kiirklahvi CtrlU (Save Default Settings (salvesta vaikeseadistus)).
- **Valgusallikas (*täpsemini Lamp tüüpi valgus*):** Vaikimisi on igas uues Blenderi stseenis üks valgusallikas (Light Source), mis asub kusagil globaalse 3D-koordinaadistiku keskpunkti lähedal.
- **Kaamera (*Camera*):** Vaikimisi on igas uues Blenderi stseenis kaamera (Camera), mis asub globaalse 3D-koordinaadistiku keskpunkti lähedal ja on suunatud selle poole.
- **Valitud objekt (*Selected object*):** Sellel väljal on näha hetkel valitud objekti nimi.

3D-akna päis

See on 3D-akna päis. Kõigil Blenderi akendel on päis (mis, nagu antud juhul, võib asuda hoopis akna allääres, aga me nimetame teda ikka päiseks).

Loe pikemalt [Blenderi päistest](#) »

Vaata selgituseks eespool olevat pilti.

- **3D-vaate varjutus (*Viewport Shading*):** Blender joonistab 3D-aknas objekte [OpenGL](#)-i abil. Sa saad määrata igas 3D-aknas kasutatava varjutustüübi, klõpsates sellele väljale ja valides ühe võimalikest varjutusviisidest. Võimalikud valikud on piirdkastist kuni täielikult tekstuuritud varjutuseni. Kui tahad kasutada tekstuuritud varjutust, peaks sul olema piisavalt võimas graafikakaart.
- **Kihid (*Layers*):** Lihtsustavad modelleerimist ja animeerimist. Blenderi kihtide abil saad jagada objekte gruppidesse vastavalt nende rollidele. Näiteks võib ühel kihil olla vesi ja teisel puud. Või siis võivad kaamerad ja valgusallikad olla omaette kihil. Et muuta stseeni selgemaks, saad sa kihte sisse ja välja lülitada.

Nuppude (omaduste) akna päis (*Buttons/Properties*)

Omaduste akna päis on ülaloodud pildil veidi tumendatud.

Vaata selgituseks eespool olevat pilti.

Nuppude (omaduste) aken sisaldab paneele, mis on omakorda grupeeritud. Akna päisel on rida nuppe (mida nimetatakse kontekstinuppudeks (*Context Buttons*)), mis võimaldavad valida, milliseid paneelide grappe näidatakse.




Nuppude akna paneelid omakorda aitavad grupeerida omavahel seotud nuppe ja seadeid. Mõned paneelid on nähtavad ainult siis, kui teatavat tüüpi objekt on valitud. Paneele saab kokku kerida, klõpsates väikesel noolel paneeli päisest vasakul (st *Render* kõrval), ja ümber paigutada, lohistades neid ülemisest paremast nurgast.

Ülevaateaken (*Outliner Window*)

Selles aknas näidatakse kõiki stseenis leiduvaid objekte. See on väga kasulik, kui töötad suurte, paljude objektidega stseenidega.

Selle akna päises saab valida, mis tüüpi elemente ja kuidas allpool esitatakse.

Ajatelg (*Timeline Window*)

Selles aknas näidatakse ajatelge, millel saab liikuda, klõpsates LMB .

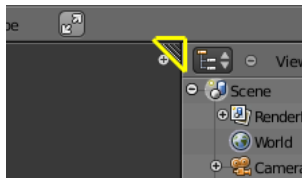
Paanide paigutamine

Blender kasutab akende organiseerimiseks omalaadset ekraani paanideks jagamist. Programmi põhiaken töölaual on alati ristkülikukujuline. Blender jagab selle mitmeks muudetava suurusega paaniks (*WindowFrame*). Paan sisaldab vastava aknatüübi, näiteks 3D-vaate (*3D View*) või Ülevaateakna (*Outliner*) tööala. Iva seisneb selles, et programmiaken on jagatud suvaliseks hulgaks väiksemateks (kuid ikka nelinurkseteks) akendeks, mis üksteist ei varja. Selle tulemusena on kõik aknad alati täielikult nähtavad ja nii on kerge liikuda tööga ühest aknast teise.

Akna maksimeerimine

Et teha käesolev paan terve Blenderi programmiakna suuruseks, vali menüüst View → Toggle Full Screen (Vaade & rarr Täisekraani lülitamine). Paani esialgse suuruse taastamiseks vali uuesti View → Toggle Full Screen. Kiirem viis paani maksimeeritud ja normaalsuuruse vahel vahetada on kiirklahvidega ⇧ ShiftSpace, Ctrl↓ või Ctrl↑. MÄRKUS: Kiirklahvide kasutamisel maksimeeritakse see aken, kus hiirekursor parasjagu on.

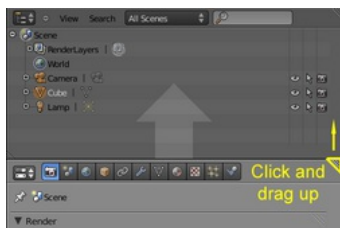
Akna jagamine



Akna ülemises paremas ja alumises vasakus ääres on aknajakajad, mis meenutavad liugurite kurrulisi sõrmepeidemeid. Nende abil saab akent nii kaheks jagada kui ka osasid jälle omavahel kokku liita. Kui sa liigutad hiire jagaja kohale, muutub kursor ristkülikuliseks. Akna vertikaalseks jagamiseks hoi a LMB ja lohista vasakule, horisontaalseks jagamiseks paremale.

Kahe paani liitmine

Et kahte paani liita, peavad nad olema liitmise suunas samade mõõtmetega. Näiteks kui tahad liita kahte kõrvuti asetsevat paani, peavad nad olema sama kõrgusega. Kui vasak paan ei ole paremaga ühekõrgune, siis neid horisontaalselt liita ei ole võimalik. Selle põhjus on see, et kombineeritud aken peab jääma ikka ristkülikukujuliseks. Sama reegel kehtib ka ülestikku asuvate paanide puhul, need peavad olema ühelaiused. Kui ülemine paan on vertikaalselt jagatud, tuleb tema osad enne selle liitmist alumise paaniga omavahel ühendada.



Et ühendada paan temast ülalpool olevaga (pildil ühendatakse omaduste paan struktuuriaknaga "viimase peale"), liiguta hiirekursor aknajakaja käepideme peale. Kui kursor muutub ristkülikuliseks, vajuta LMB ning liida aknad hiirega ülespoole lohistades. Ülemine aken muutub pisut tumedamaks ja tema peale ilmub üles suunatud noole kujutis. See märgib, et alumine (valitud) paan võtab tumendatud paani ala endale. Paanide ühendamiseks lase LMB lahti. Kui tahad ühendada vastupidises suunas, liiguta kursor tagasi algele (alumisele) paanile ja ühendusnool ilmub hoopis selle kohale.

Samal viisil saab ühendada vasakpoolse akna parempoolsega ja vastupidi.

Kui enne hiirenupu vabastamist vajutad Esc, katkestab see ühendamise.

Akna suuruse muutmine

Paani suurust saab muuta, kui vajutada raamil LMB ja hiirega lohistada. Lihtsalt liiguta hiir kahe paani vahelise raami peale, kuni kursor muutub kahesuunalise noole kujuliseks, siis klõpsa ja lohista.

Sisu vahetamine

Saad kahe paani sisud omavahel vahetada, vajutades esialgse paani ühe jagaja peal Ctrl LMB, lohistades hiirega kuni teise paanini ja lastes seal nupu lahti. Need kaks paani ei pea olema kõrvuti, kuid peavad asuma samas aknas.

Uue akna avamine

Sul võib olla soov avada uus Blenderi programmiaken, mille sees paanid asuvad. See võib näiteks olla kasulik siis, kui sul on mitu monitori ja tahad, et igaühes neist oleks näha erinev informatsioon samast Blenderi programmist.

Selleks tuleb lihtsalt vajutada aknajakajal ⇧ ShiftLMB ja veidi hiirt lohistada. Ilmub uus programmiaken koos maksimeerimis-, minimeerimis ja sulgemisnuppudega (sõltuvalt operatsioonisüsteemist), milles on üks paan. See viimane on koopia esialgsest paanist, mille jagajal sa uue akna löid.

Võid nüüd selle uue programmiakna liigutada järgmisele monitorile (või jätta olemasolevale) ja seda suurendada (või jätta suurus samaks). Saad selle sisu organiseerida kõigil eelpoolkirjeldatud viisidel (luua uusi paane ja muuta nende suurusi ning sisu) ja nii edasi.

On veel üks viis lisaakna loomiseks: File → User Preferences... (fail -> kasutaja seaded) (või CtrlAltU) avab samuti uue programmiakna, mille ainsaks paaniks on *kasutajaeelistuste* (User Preferences) aken. Saad selle programmiakna sisu samal viisil ümber korraldada.

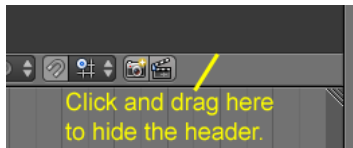
Akende päised (*Headers*)

Kõigil akendel on päis (heledama halli taustaga riba, millel on ikoonidega nupud). Mõnikord nimetame me akna päist ka tööribaks (*Toolbar*). Päis võib olla kas akna ülaosas (nagu omaduste aknal (*Properties*)) või alaosas (nagu 3D-vaate aknal (*3D Window*)). Alltoodud pildil on 3D-vaate akna päis.



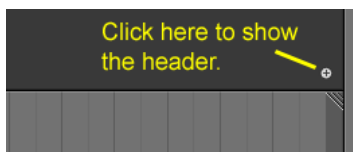
Kui liigutad hiire akna kohale, muutub päis tooni võrra heledamaks. See tähendab, et aken on "fookuses". Kõik kiirklahvid, mida vajutad, mõjuvad nüüd sellele aknale.

Päise peitmine



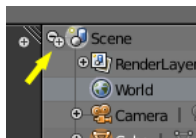
Et peita akna päist, liiguta hiir akent ja päist eraldavale joonele, kuni kursor muutub üles-alla noole kujuliseks. Siis vajuta LMB ja lohista nuppu all hoides akna suunast päise poole.

Päise nähtavaks tegemine



Peidetud päisest jääb akna nurka väike plussmärk. Klõpsa märgil LMB ja päis muutub jälle nähtavaks.

Märkus 1: 3D-vaate aknas võib olla veel kuni kaks sellist väikest plussmärki (akna ülemises vasakus ja paremas nurgas). Need avavad lisatööriistadega paneelid, mitte teise päise.



Märkus 2: Mõnedes akendes võib olla raske seda plussmärki märgata, sest ta võib tunduda ühena muudest aknas olevatest ikoonidest. Üks selline näide on struktuuriredaktor (*Outliner*), kus on veel teisi plussmärke, nii et päise taasavamise märk võib nende vahele ära kaduda.

Päise asend

Et liigutada päist ülevalt alla või vastupidi, lihtsalt klõpsa sellel RMB ja vali menüüst vajalik toiming. Kui päis on akna ülaosas, on vastava menüürea tekst "Flip to Bottom" (pööra alumiseks), ja kui päis on akna alaosas, siis "Flip to Top" (pööra ülemiseks).



Värviteemad

Blender võimaldab vastavalt kasutaja soovidele muuta enamikku kasutajaliidese värve. Kui leiad, et sinu ekraanil nähtavad kasutajaliidese värvid ei vasta sellele käsiraamatule, võib see tulla sellest, et oled muutnud vaiketeemat. Uue teema loomiseks või olemasoleva muutmiseks tuleb avada [kasutajaeelistuste](#) (*User Preferences*) aken ja valida sealt kaart Themes (teemad).

Akna tüübi nupp

Klõpsates LMB akna päise vasakpoolisel esimesel nupul, ilmub menüü, mis võimaldab valida 16 erineva aknatüübi vahel. Iga Blenderi paan võib sisaldada ükskõik millist aknatüüpi. Kui tahad kõigis paanides näha 3D-vaadet, muuda lihtsalt kõigi akende tüüpi.

Menüüd ja nupud

Anamikul aknapäistel on kohe "aknatüübi" nupu kõrval rida menüüsid, mida saab vajadusel peita, jällegi klõpsates väikesele miinusemärgile. Kui sa ei leia menüüd, millest käsiraamatule või õppetükis räägitakse, proovi leida see väike plussmärk "aknatüübi" nupu kõrval. Sellel klõpsates LMB ilmub menüü nähtavale.

Menüüd sisaldavad enamikku põhilisi valikuid ja käske, seega sirvi nad läbi ja vaata, mis võimalused olemas on. Kui tööriistal on kiirklahv, näidatakse seda ka vastaval menüüreal.

Menüüd ja nupud muutuvad vastavalt valitud aknatüübile, valitud objektile ja töörežiimile. Menüüdes on näha ainult käsud, mida on parajasti võimalik kasutada.

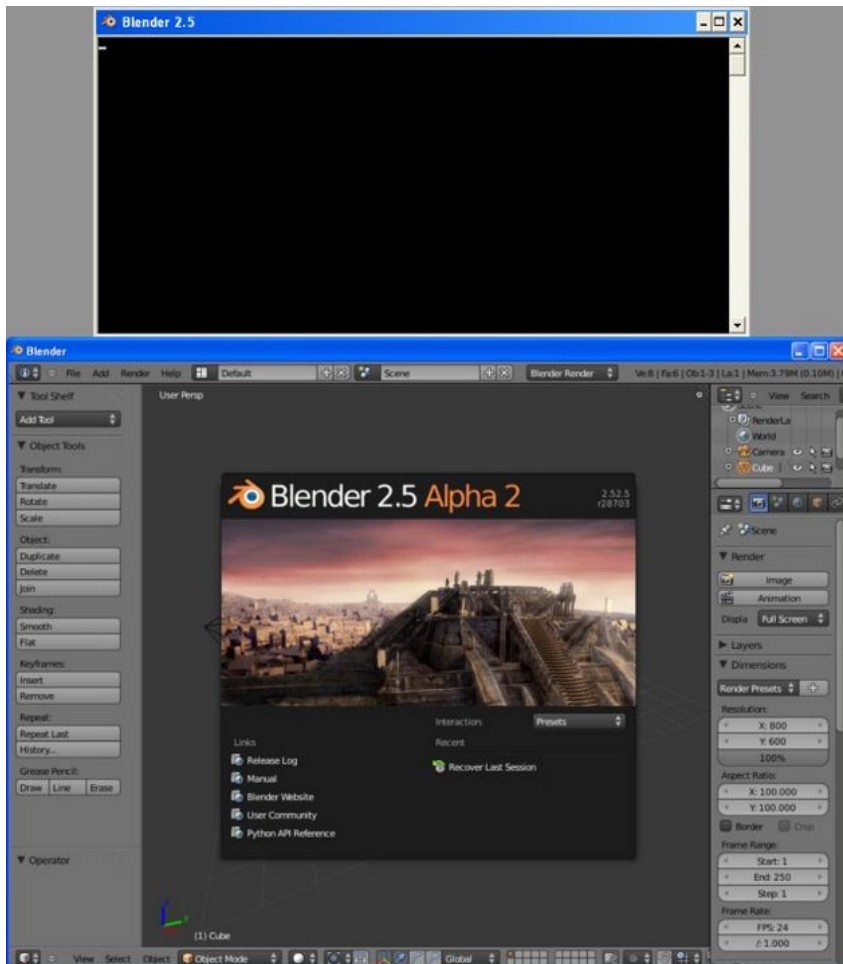
Tekstikonsooli aken (*Console Window*)

Console Window on operatsioonisüsteemi tekstiaken, mis näitab informatsiooni Blenderi toimingute, seisundi ja sisemiste vigade kohta. Kui Blender kokku jookseb, võib tekstikonsool selle põhjuste kohta lisateavet anda.

Windows 2000/XP/Vista/7

Kui käivitada Blender Windowsi operatsioonisüsteemis, avatakse tekstikonsool töölaual eraldi aknas. Kui kõik vajalikud tingimused käivitamiseks on täidetud, avaneb Blenderi põhiaken ja tekstikonsool lülitatakse välja. See erineb Blenderi versioonist 2.4x, milles jäi konsool nähtavaks kogu Blenderi töötamise aja. Selleks, et uues Blenderi versioonis tekstikonsooli näha, vali menüüst Help » Toggle System Console (abi->lülita tekstikonsool sisse/välja).

Järgneval ekraanipildil on näha kaks Blenderi akent Windows XP keskkonnas.



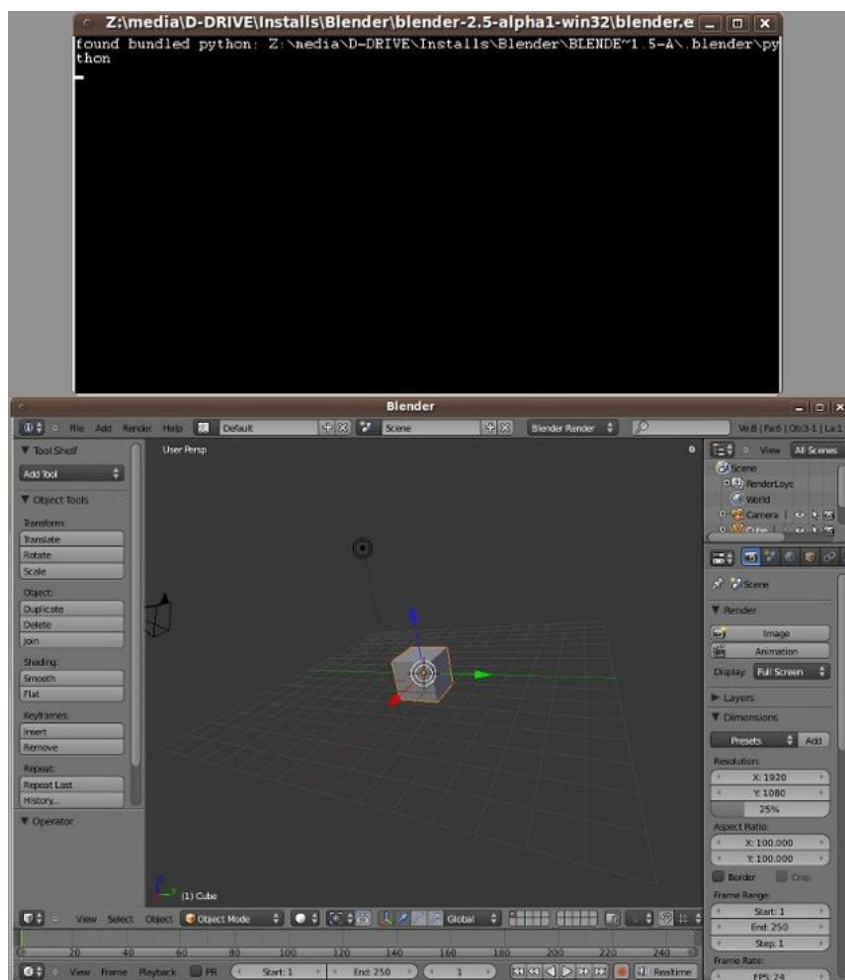
Blenderi tekstikonsool ja tööaken.

Linux

Üldjuhul on Blenderi tekstikonsool Linuxis näha ainult siis, kui Blender on käivitatud tekstiterminalist, sest Blender kasutab tekstikonsooli väljundi jaoks sedasama terminali, milles ta käivitati.

Sõltuvalt kasutatavast töökeskkonnast võib pärast Blenderi paigaldamist olla tema ikoon ka töölaual või saab Blenderi käivitada peamenüüst. Kui käivitad Blenderi töölaualikoonist või menüüst, on tema konsooliväljund tõenäoliselt peidetud sellele terminalile, millelt käivitati [XWindows](#) server.

Sellel ekraanipildil on näha Linuxi tekstiterminalilt käivitatud Blender ja sinnaasamasse väljastatud konsoolitekst.



Tekstiterminalilt käivitatud Blender Linuxis



Blenderi tekstikonsooli sulgemine

Blenderi tekstikonsooli aken peab jääma avatuks kogu Blenderi töötamise ajal. Tekstikonsooli sulgemine paneb Blenderi kinni ja a kaotad kogu salvestamata töö. MS-DOSi käsuaknad ja Blenderi tekstikonsool näevad sarnased välja — veendu alati, et sulged õige akna.

Tekstikonsooli teadaanded ja veateated

Blenderi tekstikonsoolil kuvatakse mitmesuguseid teadaandeid ja veateateid. Mõned teated annavad lihtsalt tagasisidet sellest, mida Blender parajasti teeb ja ei tähenda, et programmi töös esineks häireid. Teised teated võivad aga märkida tõsiseid vigu, mis tõenäoliselt ei võimalda Blenderil käsilolevat protseduuri lõpetada ja võivad isegi muuta Blenderi käskudele allumatuks või suisa kokku jookсутada. Blenderi tekstikonsooli teated võivad pärineda Blenderi enda koodist või välisest allikast — neist võib mainida näiteks [Pythoni skripti](#) ja [pluginaid](#).

Tavalisemad teated

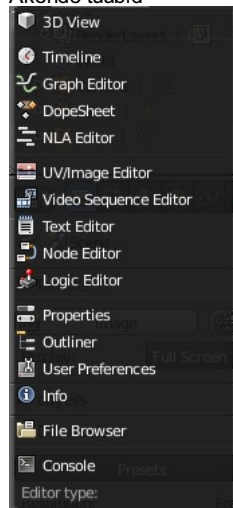
- found bundled python: (FOLDER) (leiti sisseehitatud Python: (Kaust))

See tekst annab teada, et Blender leidis sisseehitatud Pythoni interpretaatori käivitamiseks vajaliku [Pythoni](#) teegi. Kui see kaust puudub või Blender seda ei leia, tekib tõenäoliselt viga ning seda teadet ei kuvata.

- malloc returns nil() (malloc annab vastuseks NULL)

Kui Blender teostab toimingut, mis vajab täiendavat operatiivmälu (RAM), kutsub ta välja süsteemiteegi protseduuri malloc (lühend väljendist "memory allocate" (hõiva mälu)), mis üritab nõutud kogust operatiivmälu Blenderile reserveerida. Kui see pole võimalik, annab malloc tulemuseks nil/null/0, mis tähendab, et tal ei õnnestunud mälu reserveerida. Sel juhul pole Blenderil võimalik kasutaja antud käsku täita. Tõenäoliselt on muutub Blender selle tagajärjel väga aeglaseks või jookseb kokku. Kui tahad vähendada võimalust, et Blenderi kasutamise ajal vaba operatiivmälu otsa saab, võid lisada arvutile mälu, kasutada vähem ja väiksema elementide arvuga mudeleid või sulgeda Blenderiga töötamise ajal teised programmid ning teenused, mis mälust oma osa nõuavad.

Akende tüübid



Aknatüübi (Window Type) valiku menüü.

Blenderi kasutajaliides jagab operatsioonisüsteemi neljakandilise akna paljudeks väiksemateks riskülikukujulisteks paanideks (*WindowFrame*). *Igas paanis olev sisu sõltub tema aknatüübist (WindowType).*

Iga paan on teistest täiesti sõltumatu ja sa saad paigutada mitmesse paani sama tüüpi akna. Näiteks võib sul olla korraga mitu 3D-akent, millest igaüks näitab stseeni erineva nurga alt. Sa saad paane (window frame) jagada, ühendada ja nende suurust vastavalt töö iseloomule muuta. Ekraanipinna kokkuvõtmiseks saad sa määrata ka seda, kas akna päis (header) on nähtaval või mitte.

[Loe lähemalt paanide paigutamisest »](#)

Aknatüüpe jagatakse vastavalt funktsioonile:

- [3D vaade \(3D View\)](#) – sinu stseeni graafiline esitus.
- [Sõlmeredaktor \(Node Editor\)](#) – võimaldab sul kasutada tekstuuride ja materjalide loomiseks ning komposiitmiseks omavahel ühendatud sõlmi.
- [UV/pildiredaktor \(UV/Image Editor\)](#) – pildiredaktor koos võimsate UV-laotuse haldamise vahenditega.
- [Omadused \(Properties Editor\)](#) – näitab hetkel valitud objekti erinevaid omadusi.
- [Failisirvija \(File Browser\)](#) – kasutatakse failide organiseerimiseks, laadimiseks ja salvestamiseks (enamasti ilmub see aknatüüp vajadusel automaatselt).
- [Ülevaade \(Outliner\)](#) – aitab sul objekte leida ning organiseerida.
- [Ajatelg \(Timeline\)](#) – võimaldab juhtida animatsioonide esitamist.
- [Graafikuredaktor \(Graph Editor\)](#) – võimaldab orginaseerida animatsiooni võtmeid (ja juhte) ning nende interpoleerimist (ja ekstrapoleerimist).
- [NLA redaktor \(NLA Editor\)](#) – võimaldab organiseerida mittelineaarset animatsioonide jada.
- [Tööleht \(Dope Sheet\)](#) – võimaldab kombineerida üksikuid tegevusi tegevuste jadadeks.
- [Videomonteerija \(Video Sequence Editor\)](#) – võimaldab panna videoklippe kokku filmilõiguks.
- [Loogikaredaktor \(Logic Editor\)](#) – mänguloogika muutmise vahend.
- [Tekstiredaktor \(Text Editor\)](#) – võimaldab teha oma projekti kohta märkmeid ja dokumentatsiooni ning kirjutada Pythoni skripte.
- [Konsool \(Console\)](#) – võimaldab kasutada Pythonit otse Blenderis.
- [Seaded \(User Preferences\)](#) – võimaldab Blenderit sinu tööstiili ja arvuti võimalustega arvestades seadistada.

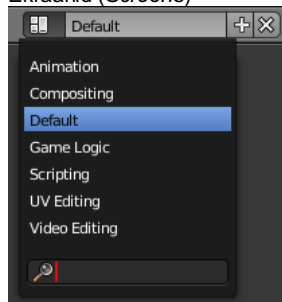
Saad määrata akna tüübi, vajutades akna päise *kõige vasakpoolsemat* nuppu. Ilmub menüü, milles kuvatakse kõik võimalikud aknatüübid (vaata pilti *Aknatüübi (WindowType) valiku menüü*).

Iga aknatüübi kohta täpsema informatsiooni saamiseks klõpsa ülaltoodud linke või vaata kokkuvõtvat peatükki [III Windows](#).

Vaata ka

- [Nupud](#)
- [Aknad](#)

Ekraanid (*Screens*)



Ekraanipaigutuse menüü

Blenderi paindlik kasutajaliides võimaldab sul luua erinevate tööprotsesside, näiteks modelleerimise, animeerimise ja tekstide/skriptide kirjutamise jaoks kohandatud ekraanipaigutusi. Sama faili piires on mõnikord kasulik töökeskkonna ülesehitust vahetada.

Iga peamise loometegevuse tüübi jaoks on Blenderil mitu eelhäälestatud *ekraani*, mis pakuvad iga töö kiirendamise ja tõhustamise tarvis oma ekraanipaigutust. *Ekraanid* on sisuliselt eelhäälestatud akende paigutused. Kui sa ei suuda leida mingit kindlat ekraani, saad kasutada nimekirja allosas asuvat otsingut (näha parempoolsel pildil).

Vaikimisi olemasolevad ekraanid

Animation (animatsioon)

Tegelaste ja teiste objektide (sealhulgas ka nende kuju, värvide jne) liikuma panemine.

Compositing (komposiitimine)

Erinevate stseeni osade (st taust, tegutsejad, eriefektid) ühendamine ja pildi filtreerimine (st värvikorrektuur)

Default (vaikimisi)

Ekraani vaikepaigutus Blenderi käivitamisel ja uue faili loomisel. Kasulik uute objektide modelleerimisel.

Game Logic (mänguloogika)

Blenderi mängude planeerimine ja programmeerimine.

Scripting (kirjutamine)

Oma töö dokumenteerimine ja/või skriptide kirjutamine Blenderi automatiseerimiseks.

UV Editing (UV lahtilõige)

Võreobjekti projektsiooni laotamine 2D-pinnale, et määrata tekstuuride paigutus objektile.

Video Editing (videotöötlus)

Animatsiooniklippide lõikamine ja muutmine.

Blender sorteerib need ekraanipaigutused nimekirjas automaatselt tähestikulises ja/või numbrilises järjekorras. Nimekiri on kättesaadav infoakna (Info Window) päisest, mis on vaikepaigutustes alati tööakna ülemisel äärel. Uued Blenderi kasutajad peavad seda tihti lihtsalt menüüribaks, kuid tegelikult on ta üks Blenderi akendest, millest on näha ainult päis.

Et valida tähestikulises nimekirjas järgmine ekraanipaigutus, vajuta Ctrl→; et valida eelmine, vajuta Ctrl←.



Ekraani ja stseeni valikud

Vaikimisi peab iga ekraanipaigutus 'meeles' viimast [stseeni \(Scene\)](#), kus teda kasutati. Uue ekraanipaigutuse valimine muudab akende paigutust ja ühtlasi hüppab sellele stseenile.

Kõik peatükkides [Aknad](#) ja [Aknatüübid](#) kirjeldatud muutused akende seadetes salvestatakse koos ekraaniga. Kui muudad aknaid ühes ekraanipaigutuses, ei mõjuta see teisi ekraane.

Ekraanide seadistamine

Uue ekraani lisamine

Vajuta nuppu "lisa" (+) ja Blender loob uue ekraanipaigutuse, mis on hetkel kasutatava ekraani koopia.

On mõistlik anda uuele ekraanile mitte ainult *nimi*, vaid ka *number* selle ees, et sa teaksid nooleklahvidega ekraanide vahetamisel nende täpset järjekorda. Saad ekraanipaigutuse ümber nimetada, klõpsates nimeväljal LMB (☞) ja kirjutades uue nime või siis klõpsates veel kord – mis paigutab väljale kursori ja laseb seda muuta. Näiteks võid sa kasutada nime "6-MyScreen". Vaata pilti *Ekraani ja stseeni valikud* ülalpool.

Ekraani kustutamine

Saad ekraani kustutada, vajutades nuppu [Template:LiteralDelete Datablock](#) (☒) – kustutab andmebloki). Vaata pilti *Ekraani ja stseeni valikud* ülalpool.

Ekraani ümberkorraldamine

[Käepidemete ja raamide](#) abil saad muuta paanide paigutust ning paane vastavalt vajadusele jagada ja liita. Kui oled paigutusega rahul, vajuta CtrlU, et oma vaikeseadistusi uuendada. Ära unusta, et kasutaja seadetes kirjutatakse ka stseenid, seega tasub sul ekraanipaigutusi häälestada ainult ühe vaigestseeniga.

Omaduste aknas (Properties) on spetsiaalne valik: klõpsates selle taustal RMB (☞), saad paigutada paneelid kas horisontaalselt või vertikaalselt. Nendest kahest valikust on vertikaalselt paigutatud paneelid parema toega.

Vaikeseadete üle kirjutamine

Kui salvestad .blend-faili, salvestatakse ekraanipaigutused koos sellega. Faili avamisel märgib linnuke Load UI (loe kasutajaliides) failisirvijas seda, et Blender loeb ekraanipaigutused failist (ja kirjutab selle käigus sinu vaikeseaded üle). Kui jätta linnuke Load UI (loe kasutajaliides) tühjaks, kasutab Blender parasjagu kehtivat ekraanipaigutust.

Täiendavad paigutused

Kui sinu kogemused Blenderis kasvavad, tasub sul ilmselt lisada mõned täiendavad ekraanipaigutused, mis sobivad sinu enda tööstiiliga ja tõstavad nii töö produktiivsust. Mõned näited võiksid olla:

1-Model (modelleerimine)

Neli 3D-akent (vaated ülalt, eest, küljelt ja perspektiivis), omaduste aken objektide muutmiseks

2-Lighting (valgustus)

3D-aknad valgusallikate liigutamiseks, UV/pildiredaktor renderdatud pildi näitamiseks, omaduste aken valgusallikate ja renderduse seadistamiseks

3-Material (materjal)

Omaduste aken materjali omaduste muutmiseks, 3D-aken objektide valimiseks, ülevaateaken (Outliner, *Library script* (vajadusel), sõlmeredaktor (kui kasutad [sõlmmaterjale](#)))

4-UV Layout (UV laotus)

UV/pildiredaktor, 3D-vaade õmbluste tegemiseks ja lahtilõikeks

5-Painting (maalimine)

UV/pildiredaktor tekstuuride maalimiseks pildile, 3D-aken tekstuuri maalimiseks otse objektile, külje peal kolm mini-3D-akent, milles on määratud maksimaalse tugevusega taustapilt, omaduste aken

6-Animation (animatsioon)

Graafiku redaktor, 3D-aken skeleti pooside seadmiseks, NLA aken

7-Node (sõlmed)

Suur sõlmeredaktor sõlmekaardi koostamiseks, UV/pildiredaktor, mis näitab renderdatud tulemust

8-Sequence (montaaž)

Graafikuredaktor, videomonteerija ülevaatepildi režiimis, videomonteerija ajatelje režiimis, ajatelg ja vana hea omaduste aken.

9-Notes/Scripting (märkmekd/skriptid)

Ülevaateaken, tekstiredaktor

Taaskasuta paigutusi

Kui sa lood uue paigutuse, mida tahad tulevikus teistes .blend-failides kasutada, salvesta ta lihtsalt kasutaja vaikeseadistustesse, vajutades CtrlU (ära unusta, et salvestatakse kõik ekraanid ja samuti stseenid).

Stseenid

Stseenid on praktiline viis oma projektide organiseerimiseks. Kuup kaamera ja lambiga, mida näed Blenderi esmakordsel käivitamisel, on vaigestseen. Võid kujutada stseene ette nagu veebilehitseja kaarte (tabs). Näiteks võib veebilehitsejas olla korraga avatud palju kaarte. Kaardid võivad olla tühjad, neis võivad olla avatud sama veebilehe identsed kohad, sama veebilehe erinevad kohad või täiesti erinevad lehed. Blenderi stseenid töötavad suuresti samal viisil. Sul võib olla täiesti tühi stseen, täiesti identne koopia mõnest teisest stseenist või uus koopia, mis on algse stseeniga mitmel erineval viisil seostatud (lingitud).

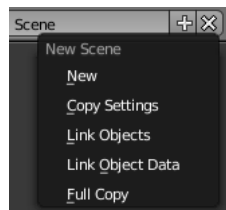
Saad stseene valida ja uusi luua stseenivalija (Scene Selector) abil infoakna päisel (enamike Blenderi ekraanipaigutuste kõige ülemine riba - vaata pilti *Ekraani ja stseeni valikud*).




Ekraani ja stseeni valikud

Stseeni seadistamine

Uue stseeni lisamine



Stseeni lisamise menüü

Uue stseeni saad lisada, vajutades nuppu  stseenivaliku väljal. Uue stseeni loomisel saad tema algse sisu valida viie võimaluse hulgast (pilt *Stseeni lisamise menüü*).

Et nende võimaluste vahel valida, pead sa selgelt mõistma vahet "objektide" (*Objects*) ja "objektiandmete" (*Object Data*) vahel. Iga Blenderi graafiline element (võre, valgusallikas, kõver *jne*) koosneb kahest osast: objektist (*Object*) ja objektiandmetest (*Object Data*) (mõnikord kasutatakse ka lühendit *ObData*). Objekti on määratud selle üksiku elemendi asukoht, pööre ja suurus. Objektiandmed sisaldavad informatsiooni võre, materjalide jms kohta. Objektiandmed on ühised kõigile selle objekti kloonidele (*Instance*). Iga objekt sisaldab viidet (linki) oma objektiandmetele. Ühtesid ja samu objektiandmeid võivad jagada paljud objektid.

Need viis valikut määravad seega, kui palju informatsiooni *kopeeritakse* parasjagu valitud stseenist uude ja kui palju *jagatakse* ("lingitakse"):

New (uus)

Loob tühja stseeni. Selles uues stseenis on renderdamise seadetel samuti vaikeväärtused.


Copy Settings (kopeeri seaded)

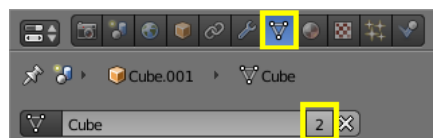
Loob uue stseeni nagu eelmine valik, kuid kopeerib renderdamise seaded olemasolevast stseenist uude.

Link Objects (lingi objektid)

Kõige "nõrgem" stseeni kopeerimise variant. See loob uue stseeni täpselt sama sisuga, nagu oli eelmine stseen. Kuid objektide kopeerimise asemel need *lingitakse* (seostatakse) objektitasemel eelmise stseeni objektidega. Seega kõik muudatused *uues* stseenis muudavad täpselt samamoodi *algse* stseeni objekte, sest uue stseeni objektid on *päriselt* needsamad, mis vana stseeni omad. Vastupidine kehtib samuti (muudatused *vanas* stseenis muudavad samamoodi *uut* stseeni).

Link Object Data (lingi objektiandmed)

Loob uued koopiad (duplikaadid) käesoleva stseeni objektidest, kuid igaüks neist duplikaatobjektidest on *lingitud* (seostatud) samade objektiandmetega (võred, materjalid ja nii edasi), nagu vastavad objektid algse stseenis. See tähendab, et saad muuta uue stseeni objektide asukohta, asendit ja suurust ilma teiste stseenide objekte muutmata, kuid kõik objektiandmed (võred, materjalid *jne*) muudatused kajastuvad ka teistes stseenides. See on nii seepärast, et kõigi sel viisil seostatud stseenide vastavad objektid jagavad *ühte objektiandmete komplekti*. Kui tahad muuta objekti *uues stseenis* ilma teiste stseenide objekte mõjutamata, pead sa selle objekti tegema *uues stseenis* "ühe kasutajaga" (*single-user*) koopiaks, klõpsates LMB  koopiade numbri väljal *omaduste akna* objektiandmete paneelil (*Object Data*). Sellega luuakse vastava objekti jaoks objektiandmetest uus iseseisev koopia.



Full Copy (täielik koopia)

See on kõige täielikum kopeerimise variant. Stseenide vahel ei jagata mitte midagi. Selle valikuga luuakse täiesti iseseisev stseen, mille sisu on koopia käesoleva stseeni sisust. Igast algse stseeni objektist luuakse duplikaat ja lisaks tehakse talle ka isiklik, privaatne koopia objektiandmetest.

Et Blenderi andmehaldust paremini mõista, loe peatükki [Blenderi teegid ja andmesüsteem](#).

Lühike näide

Mõtle baaristseenile filmis. Lood alguses puhta versiooni baarist, kus kõik esemed on terved ja oma kohtadel. Siis otsustad tegevused teises stseenis teha. Tegevuse iseloom määrab, mis tüüpi linkimine (kui üldse) sobib antud stseenile kõige paremini.

Link Objects (lingi objektid)

Iga objekt lingitakse algse stseeniga. Kui parandad seina asendit, muutub see ka igas teises stseenis, mille tegevuspaigaks on seesama baar.

Link Object Data (lingi objektiandmed)


On kasulik siis, kui objektide asend peab muutuma, aga nende kuju ja materjal jäävad samasugusteks. Näiteks võivad toolid olla "rahvast täis" baaristseenis põrandal ja "sulgemisagses" stseenis laudadele tõstetud. Toolid ei muuda oma kuju ja seetõttu pole vaja raisata mälu nendest täpsete koopiade tegemisele.

Full Copy (täielik koopia)

Põrandale kildudeks kukkuv klaas nõuab isiklikku koopiat objektiandmetest, sest ta muudab kuju.

Kõike eelnevat ei saa ühes ja samas stseenis teha, kuid see võib siiski aidata mõista, millal ja miks linkida objekte eri viisidel.

Stseeni kustutamine

Saad stseeni kustutada, vajutades nuppu Delete Datablock (kustuta objektiandmed) () stseenivalijas (vaata pilti *Ekraani ja stseeni valikud*).

Kontekstid (*Contexts*)

Omaduste (või nuppude) aken (Properties Window) hoiab hulka erinevaid kontekste (Context), mille vahel saab valida päisel asuva ikoonirea abil (vaata pilti *Kontekstinuppude näide*).



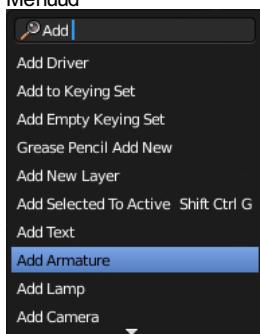
Kontekstinuppude näide

Nuppude arv ja tüübid muutuvad sõltuvalt valitud objektist nii, et näidataks ainult sisulist tähendust omavaid nuppe. Nende nuppude järjekord on hierarhiline, nagu allpool kirjeldatud:


- **Renderdamine (Render)**: Kõik, mis puudutab renderdamise väljundit (mõõtmed, sakkide silumine, kiirus jne.).
- **Stseen (Scene)**: Gravitatsioon, ühikud ja teised üldised seaded.
 - **Maailm (World)**: Keskkonnavalgustus, taevast, udu, tähed ja kaudvari (*Ambient Occlusion*).
 - **Objekt (Object)**: Teisendused, esituse seaded, nähtavuse seaded (kihtide abil), duplitseerimise seaded ja animatsiooni andmed (objekti asendisse puutuvad).
 - **Piirajad (Constraints)**: Kasutatakse objekti teisenduste (asend, pööre, skaala) juhtimiseks, teise objekti või trajektoori järgimiseks ja objektidevaheliste suhete määramiseks.
 - **Töötled (Modifiers)**: Operatsioonid, mis algse objekti geomeetria muutmata muudavad tema väljanägemist renderdusel ja 3D-vaates (näiteks peegeldus ja silumine).
 - **Objektiandmed (Object Data)**: Sisaldab objektispetsiifilist informatsiooni (valgusallika värv, kaamera fookuskaugus, tipugrupid jne). Selle ikooni väljanägemine sõltub valitud objektist (siin on näidatud võreobjekti ikooni).
 - **Materjalid (Materials)**: Pinna väljanägemine (värv, läige, läbipaistvus jne).
 - **Tekstuurid (Textures)**: Materjalid kasutavad neid täiendavate detailide allikana (näiteks värv, läbipaistvus, näiline kolmemõõtmeline sügavus).
 - **Osakesed (Particles)**: Lisab objektile hulga (tavaliselt väikesi) lisaobjekte nagu valgusallikaid või võreobjekte, mida saab juhtida jõuväljade (Force Fields) abil ja ka muudel viisidel.
 - **Füüsika (Physics)**: Riide, jõuväljade, kokkupõrgete, vedelike ja suitsu simulatsiooni seaded.


Iga konteksti [nupud](#) on grupeeritud [paneelidele](#).

Menüüd



Space-menüü

Blenderis on palju menüüsid, mis on kättesaadavad kas akna päiselt või [kiirklahvi](#) kasutades otse hiirekursori asukohalt või siis vajutades RMB  akna raamil, nupul või mõnel muul kasutajaliidese elemendil. Viimasel juhul kuvatakse olukorrale vastav menüü, juhul kui see on vastavale elemendile olemas.

Lisaks saab vajutades klahvi Space avada menüü kõigi Blenderi käskudega (näidatud pildil). Lihsalt hakka trükkima vajaliku käsu nime ja automaatne otsingufunktsioon teeb sinu eest ülejäänud. Kui võimalike käskude nimekiri on piisavalt ahtaks kahandatud, klõpsa soovitud käsul LMB  või vali see klahvidega ↓ ja ↑ ning käivita, vajutades Return.

Kui tunned puudust versioon 2.4x vanast tööriistakastist, võid sa midagi sarnast lisada 3D View: Dynamic Spacebar Menu (3D-vaade: dünaamiline tühikumenüü) lisamooduli abil, mille saab paigaldada {{Literal|eelistuste (Preferences) akna} kaardilt Add-Ons (lisad).

[Loe lisadest lähemalt »](#)

Osad menüüd on kontekstitundlikud ja seega kättesaadavad ainult teatud situatsioonides. Näiteks lisavõimaluste (*Specials*) menüü (kiirklahv W) on kättesaadav ainult 3D-aknas muutmisrežiimis (*Edit Mode*).

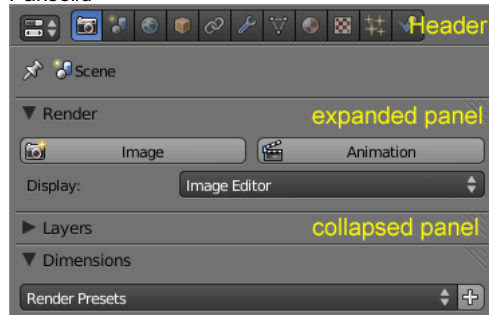
Kui sa Blenderit kasutad, jälgi alati, mis režiim on aktiivne ja mis tüüpi objekt on valitud. See aitab mõista, millised kiirklahvid millal töötavad.



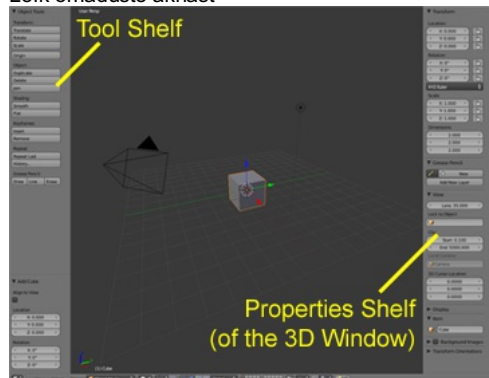
Menüüd Macil

Blender ei kasuta standardset operatsioonisüsteemi menüüd. Kui kasutad Maci, on sul ekraani ülaseras tõenäoliselt üleliigne menüüriba. Et seda eemaldada, vaata [seda postitust](#) Macworldis, kuid võta teadmiseks, et see on üpris keerukas. Alternatiiv: lihtsalt pane Blenderi tööaken infoakna päise (enamasti on see tööakna kõige ülemine äär) viimast nuppu vajutades täisekraanil töötama.

Paneelid



Lõik omaduste aknast



3D-akna paneelid

Paneelid asuvad üldiselt omaduste aknas (Properties) (Versiooni 2.4x nuppude (*Buttons*) aken), mille võib ekraani vaikepaigutuse korral leida parempoolselt servalt (vaata pilti *Lõik omaduste aknast*).

Paneelid asuvad ka tööriistariiulil (Tool Shelf) ja omaduste paneelil (Properties), mis on 3D-akna sisse ja välja lülitatavad alad. Et näha tööriistariiulit, vali menüüst View » Tool (vaade -> tööriistad) või vajuta kiirklahvi T. Et näha omaduste riulit, vali menüüst View » Properties (vaade -> omadused) või vajuta kiirklahvi N. Vaata pilti *3D-akna paneelid*.

Omaduste akna päisel on võimalik valida palju erinevaid [kontekste](#). Igal kontekstil (Context) on erinev arv ja erinevat tüüpi paneelid. Näiteks on renderdamiskontekstis (Render Context) paneelid, mis võimaldavad muuta väljundi suurust ning sakkide silumist (*antialias*), samas kui materjalikontekstis (Materials Context) on paneelid, mis võimaldavad määrata värvi, läbipaistvust, tekstuuri jne.

Omaduste akna paneele saab reastada kas horisontaalselt või vertikaalselt, klõpsates omaduste aknas RMB ja valides menüüst soovitud paigutuse. Pane tähele, et omaduste akna paneelid on optimeeritud vertikaalpaigutuse jaoks. Horisontaalne paigutus võib olla töötamiseks ebamugavam.

Paneelide paigutust ja nähtavust saab vastavalt soovile muuta. Näiteks saab paneele:

- aknas (või riilil) ringi liigutada, vajutades LMB ja lohistades ülemises paremas nurgas asuvat käepidet (see meenutab paanide jagaja käepidet ja kujutab kolmnurka).
- kerida üles ja alla, kasutades hiire ratast Wheel .
- suurendada ja vähendada, vajutades Ctrl MMB ja liigutades hiirt paremale või vasakule
- kokku kerida ja taas nähtavaks teha, klõpsates LMB paneeli päise vasakus servas asuval mustal kolmnurgal.

Täpsemalt loe iga paneeli kohta [paneelide](#) ülevaatest või leia käsiraamatust vastav peatükk.


Nupud ja juhtelemendid

Nuppe ja muid juhtelemente leiab peaaegu igas Blenderi kasutajaliidese [aknas](#). Allpool kirjeldatakse erinevaid juhtelemente.

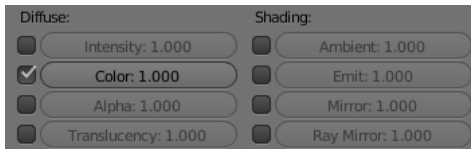
Toimingunupp



Toimingunupp

Kui nendel nuppudel klõpsata (LMB , teostavad nad mingi toimingu (nagu nupud ikka). Blenderi vaikevärvipaleti korral tunneb need ära halli värvi järgi.

Lülisnupud



Lülisnupud

Lülisnupud tunnevad ära linnukesega kasti järgi. Seda tüüpi nupule vajutamine ei teosta toimingut, vaid ainult lülitab mingi seade sisse või välja. Mõnedel juhtudel on nende nuppudega liidetud numbrisisesustaväli, mis määrab vastava omaduse mõju ulatuse.

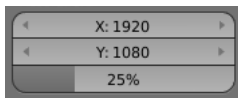
Valikunupud



Valikunupud




Valikunupud on grupp üksteist välistavaid lülisnuppe. Neid kasutatakse selleks, et valida väikese arvu "üksteist välistavate" valikute vahel.

Numbrinupp



Numbrisisesustaväljad

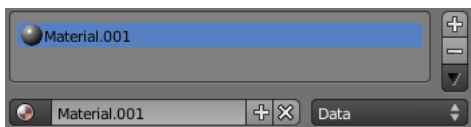
Numbrinupu tunneb ära nende teksti järgi, mis enamikul juhtudel koosneb kooloniga lõppevast nimest ja sellele järgnevast numbrist. Numbrinupul olevat väärtust saab muuta mitmel viisil:

1. Selleks, et muuta väärtust sammhaaval, klõpsa LMB  väikestel kolmnurkadel nupu külgedel.
1. Selleks, et muuta väärtust suuremas ulatuses, klõpsa LMB  ja lohista hiirega vasakule või paremale. Kui sa hoiad peale hiireklahvi vajutamist all klahvi Ctrl, muutub väärtus suurte sammudega; hoides all klahvi ⇧ Shift, on väärtuse muutus aeglasem ja juhtimine seega täpsem.
1. Vajutades klahvi ↵ Enter või LMB , saad sa väärtuse käsitsi sisestada.

Käsitsi väärtusi sisestades saad liigutada kursor välja algusesse vajutades Ctrl←, lõppu liigutamiseks vajuta Ctrl→. Muutmise katkestamiseks vajuta klahvi Esc. Võid mugavalt kopeerida numbrinupu väärtuse puhvrissi, selleks liiguta hiir nupu kohale ja vajuta CtrlC. Samamoodi saad sa kopeeritud teksti väljale kleepida kombinatsiooniga CtrlV.

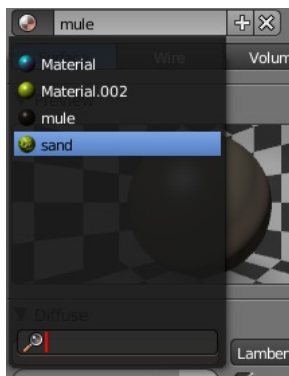
Võid sisestada ka lihtsaid avaldise, nagu näiteks 3*2 (6 asemel). Ära unusta sisestada kümnendkohtade eraldajat (ära muretse sisestatud koma pärast, see muutub ise punktiks), kui sa tahad murdarvulist tulemust (näiteks $3 / 2 = 1$, aga $3.0 / 2 = 1.5$). On võimalik kasutada isegi konstante nagu pi (3.142) ja funktsioone nagu sqrt(2) (ruutjuur 2-st).

Menüünupud



Andmebloki lingi nupud

Menüünupud võimaldavad valida üht elementi jooksvalt loodavatest nimekirjadest. Menüünuppe kasutatakse põhiliselt andmeblokkide (*Datablocks*) üksteisega linkimiseks. (Andmeblokid on objektiandmeid sisaldavad struktuurid nagu võred (*Meshes*), objektid (*Objects*), materjalid (*Materials*), tekstuurid (*Textures*) ja nii edasi. Linkides objekti materjaliga, määrad sa objektile selle materjali)



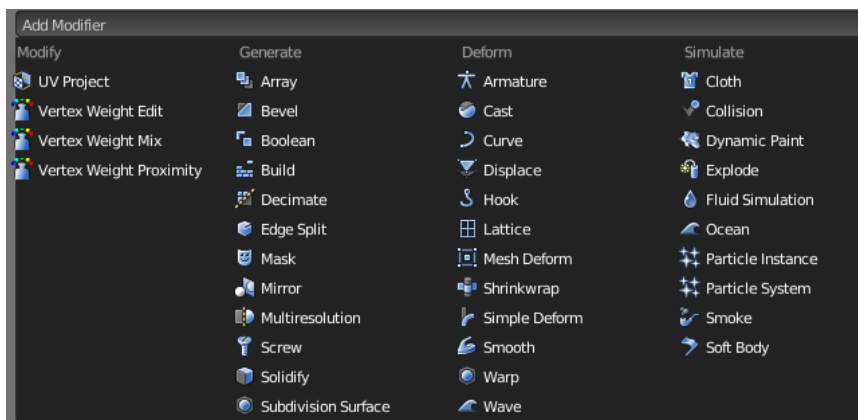
Andmebloki lingi nupud koos otsinguga

1. Esimene nupp (andmebloki tüübi ikooniga) avab menüü, mille abil saad valida linkimiseks andmebloki, klõpsates vastaval real LMB . (Nimekirja lõpus on otsinguväli.)
1. Teine nupp näitab lingitud andmebloki nime ja võimaldab sul seda muuta (klõpsates LMB nimeväljal).
1. Nupp "+" loob koopia hetkel valitud andmeblokist ja lingib selle.
2. Nupp "X" kustutab lingi.

Mõnikord on mõnel elemendil nimekirja lingitud andmeblokkidest (nagu objekti poolt kasutatavate materjalide nimekirja). Vaata pilti (*Andmebloki lingi nupud*) ülal.

1. Et valida ühte andmeblokki, klõpsa selle nimel LMB .
2. Et lisada nimekirja uut elementi (näiteks materjali, osakeste süsteemi jne) klõpsa LMB nupul "+" nimekirjast paremal.
3. Et elementi eemaldada, klõpsa LMB nupul "-" nimekirjast paremal.

Teine menüünuppude tüüp näitab staatilist valikute nimekirja. Näiteks töötleja lisamise nupp (*Add Modifier*) tekitab menüü, kus on üles loetud kõik võimalikud töötlejad.



Andmebloki lingi nupud

Linkimata objektid

Linkimata andmeid **ei kustutata enne, kui Blenderist väljud**. See on võimas omadus tehtud muudatuste taastamiseks. Kui objekti kustutad, siis tema poolt kasutatud materjal lingitakse lahti, kuid ta jääb Blenderisse alles! Pead ta lihtsalt uuesti teise objektiga linkima või looma talle "võltskasutaja" (*Fake User*) (st klõpsama Ülevaateaknas (Outliner) andmeblokkide režiimis selle andmebloki vastaval nupul).

[Loe võltskasutaja kohta lähemalt »](#)

Värvivalija

Mõned valikud avavad dialoogipaneeli. Näiteks kui klõpsata värvikastil, avaneb värvivalija (Color Selector) dialoog; vaata pilti (*Värvivalija*).



Värvivalija

Peidetud nupud

Aeg-ajalt lisavad mõned nupud paneelile täiendavaid nuppe. Näiteks paneelil Ramps (värvüleminekud) on peidetud nupp (*Cascade button*) nimega Ramp (värvisulatus), millele klõpsates ilmuvad täiendavad nupud, mis määravad värvisulatuse; vaata pilte (*Värvisulatus enne*) ja (*Värvisulatus pärast*).



Värvisulatus enne



Värvisulatus pärast

Renderdamine

Mis on renderdamine?

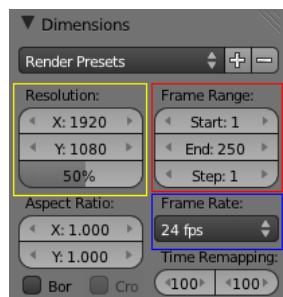
Renderdamine on kahemõõtmelise pildi loomine. Blender loob selle pildi, võttes arvesse sinu mudelit ning kõiki sinu materjale, tekstuure, valgustust ja komposiitimist.

Pildi renderdamine

Renderdamise alustamiseks võid:

- Vajutada kiirvalikut F12
- Minna paneeli Properties Window » Render kontekst » Render paneel ja vajutada pildi nuppu Image või
- Valida infoakna (Info Window) päises olevast menüüst Render » Render Image

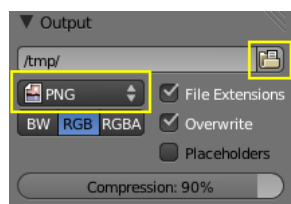
Renderduse katkestamiseks vajuta Esc.



Mõõtmete paneel

Pildilahutuse määramine

Renderduse konteksti Render mõõtmete paneeli Dimensions kaudu saad muuta pildilahutust. Vaikimisi säteteks on 50% mõõtmetest 1920 x 1080. Suuremad pildilahutused ja protsendid näitavad rohkem detaile, kuid nende renderdamine kestab kauem.



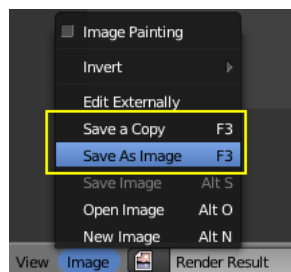
Väljundi paneel

Väljundi formaat ja väljundfail

Saad valida ka väljundiformaadi ja renderdatud pildi või animatsiooni asukoha. Vaikimisi salvestatakse nad kausta /tmp/ või /TPW/, kuid saad seda muuta vajutades väljundi paneelis Output olevale kataloogi ikoonile. Samuti saad menüünupust valida oma tööle vastava pildi- või videoformaadi.

Pildi salvestamine

Blender ei salvesta pilti automaatselt. Et pilti salvestada, võid vajutada F3 või valida UV/pildiredaktori päises olevast pildimenüüst Image sissekande 'Save as'.



Salvestamise dialoog

Animatsiooni renderdamine

Animatsiooni saad renderdada:

- Vajutades CtrlF12
- Minnes paneeli Properties Window » Render kontekst » Render paneel ja vajutades animatsiooni nupule Animation või
- Valides infoakna (Info Window) päises olevast menüüst Render » Render Animation

Renderduse katkestamiseks vajuta Esc.

Ekraanipildid ja ekraanilindistus

Et hõlbustada meeskonnatööd ja kiiret prototüüpide loomist, võib sul tekkida soov oma tööst kähku ekraanipilt võtta või seda lindistada.

Ekraanipildid

Kiirklahv CtrlF3 teeb ekraanipildi sinu Blenderi aknast ning avab seejärel failisirvija akna File Browser, kust saad määrata tehtud pildi nime ja asukoha.

Ekraanilindistus

Kiirklahv AltF3 alustab ekraanilindistust. Niimoodi lindistatakse aja jooksul sinu tegevused kas video või pildijadana. Väljundi tüüp ja asukoht määratakse [renderdamise konteksti](#) akna [väljundi paneelist](#).

Vaikimisi säte loob ekraanilindistuse, mis koosneb iga 50 ms tagant tehtud PNG piltidest ning see salvestatakse kataloogi /tmp. Kui soovid lindistada videot, määra väljundi paneelis Output oleva formaadi välja väärtuseks üks sinu süsteemi poolt toetatavaist videofaili formaatidest. Kui sa pole kindel, milliseid koodekeid sinu süsteem toetab, vali AVI JPEG. Ekraanilindistuse kaadrisageduse ja pildijada piltide tegemise vahe saad määrata [kasutajaeelistuste](#) akna [[Doc:ET/2.5/Manual/Interface/Configuration/System|süsteemi paneelist].

Ekraanipilt operatsioonisüsteemi vahenditega

Pildi saad lõikepuhvrisse ka operatsioonisüsteemi vahendite abil. Seejärel saad asetada selle pildi lõikelaualt oma pildiredaktorisse.

Ekraanipilt Windowsis

Ekraanipildi tegemiseks lõikepuhvrisse vajuta kiirklahvi CtrlPrtsr.

Ekraanipilt Mac OsX's

Vajuta kiirklahvi ⌘ Cmd+ Shift3, et salvestada ekraanipilt faili töölauale.

Vajuta kiirklahvi ⌘ Cmd+ Shift3, et kopeerida ekraanipilt lõikelauale.

Vajuta kiirklahvi ⌘ Cmd+ Shift4, et salvestada piirkond ekraanilt faili töölaual.

Vajuta kiirklahvi ⌘ Cmd+ Shift4, et kopeerida piirkond ekraanilt lõikelauale.

Ekraanipilt GNU/Linuxis

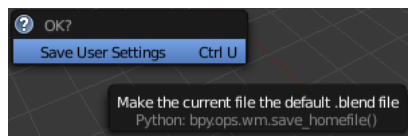
Kasutajasõbralikes süsteemides nagu Ubuntu võid vajutada kiirklahvi Prtsr, et salvestada ekraanipilt faili.

Et salvestada oma töö videona, kasuta programmi *gtk-recordmydesktop*, mis on saadaval ametlikes Ubuntu programmikogudes.


Vaikestseeni määramine

Kui käivitad Blenderi või alustad uut projekti, valides menüüst File » New (fail -> uus) või vajutades kiirklahvi Ctrl+N, luuakse uus stseen seadistustes (User Preferences) määratud vaikestseeni järgi.

Kiirklahv



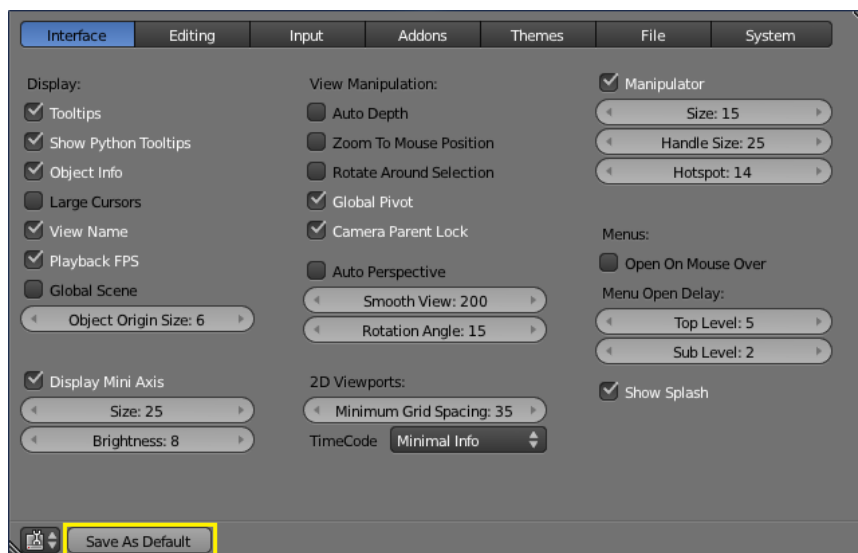
Selleks, et määrata uus vaikestseen, tee käsilolevasse stseeni kõik soovitud muudatused ja vajuta kiirklahvi Ctrl+U.

Ilmub kinnitusdialogu Save User Settings (salvesta kasutaja seaded). Klõpsa LMB  väljal Save User Settings või vajuta klahvi ↵ Enter. Katkestamiseks vajuta Esc.

Menüüst

Selleks, et määrata uus vaikestseen, tee käsilolevasse stseeni kõik soovitud muudatused ja vali menüüst File » User Preferences (fail -> seaded). Avaneb seadistuste aken (User Preferences).

Muudatuste salvestamiseks klõpsa nupul Save As Default [Template:Shortcutmb](#) (salvesta vaikeseadetena). Selleks, et jätta vaikestseen (Default Scene) muutmata, sulge seadistuste aken (nupule Save As Default (salvesta vaikeseadetena) vajutamata).



Tulemus

Käesolev stseen, kõik objektid ja seadistused salvestatakse vaikeseadistustena.

Blenderi standardse vaikestseeni taastamine

Selleks, et taastada Blenderi standardne vaikestseen, vali menüüst File » Load Factory Settings (fail -> taasta tehase seaded). See taastab paigaldamisjärgsed "tehaseseaded".

Vigadest üle saamine

Blenderis on mitu võimalust ja sätet, millega saad vähendada võimalust, et oma töö kaotad. Nende hulgas on tühistamise ajalugu (Undo History), automaatne salvestamine (Auto Save) ja lahkumisel salvestamine (Save on Quit). Kõik need võimalused on vaikimisi sisse lülitatud. Tühistamise ajalugu Undo History salvestab su tegevused nimekirja, automaatne salvestamine Auto Save pakub mitu erinevat võimalust su tööst erinevate versioonide salvestamiseks ning lahkumisel salvestamine Save on Quit salvestab Blenderi sulgemisel automaatselt faili.

Tühistamine (*Undo*)

Mode: Kõik režiimid

Hotkey: CtrlZ

Nagu enamustes programmides, vajuta viimase tegevuse tagasivõtmiseks CtrlZ.

Ennistamine (*Redo*)

Mode: Kõik režiimid

Hotkey: ⇧ ShiftCtrlZ

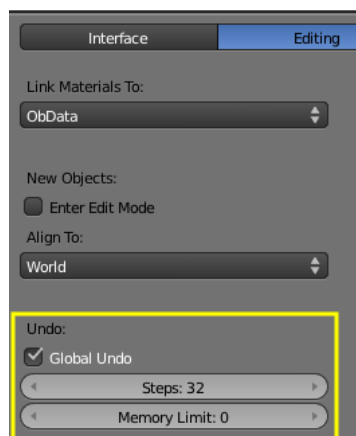
Viimati tühistatud tegevuse ennistamiseks vajuta ⇧ ShiftCtrlZ.

Tühistamise ajalugu (*Undo history*)

Mode: Kõik režiimid

Hotkey: CtrlAltZ

Blender peab meeles sinu 32 viimast tegevust, mida saad kiirvaliku CtrlAltZ abil vaadata. Ükskõik millisele tegevusele vajutamisel viiakse sind selleni tagasi tühistades kõik järgnevad tegevused. Pea meeles, et eksisteerib *kaks* eraldi ajalugu: üks mis on seotud muutmisrežiimiga Edit nin kõigi teiste režiimide jaoks mõeldud globaalne tühistamise nimekirja Global Undo. Aga sinu tegevuste meeles pidamine tuleb süsteemi mälu arvelt.



Tühistamise sätted

Tühistamise sisse- või väljalülitamiseks mine kasutaja eelistuste aknasse User Preferences ja vajuta muutmise sakile Editing. Selles sektsioonis saad määrata:

Sammud (Steps)

See numbriväli määrab mitu sammu ehk tegevust salvestatakse. Vaikimisi väärtus **32** laseb sul tühistada viimased kolmkümmend kaks tegevust, mis sa läbiviisid.

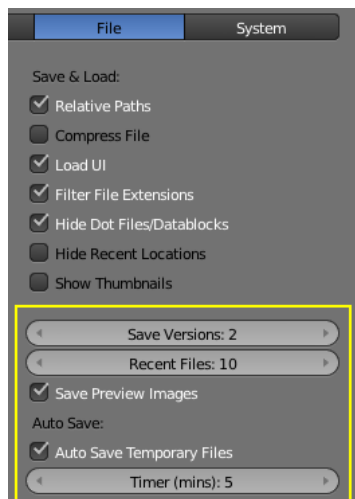
Mälu piir (Memory Limit)

See väli määrab maksimaalse mälu hulga (megabaitides), mida tühistamissüsteem tohib kasutada. Vaikimisi väärtus **0** tähistab limiidi puudumist.

Globaalne tühistamine (Global Undo)

Lubab Blenderil salvestada tegevusi, mida tehti **väljaspool** muutmisrežiimi Edit. Näiteks objektide dubleerimist, paneeli sätete muutmist või režiimide vahel liikumist.

Salvestamine ja automaatne salvestamine



Automaatse salvestamise sätted

Arvuti kokku jooksmised, voolu äraminekud või lihtsalt salvestamise unustamine võivad põhjustada sinu töö kadumise või kasutuskõlbmatuks muutumise. Selle võimaluse vähendamiseks saab Blender kasutada automaatset salvestamist Autosave. Kasutajaeelistuste akna User Preferences faili sakist File saad konfigureerida need kaks võimalust, mida Blender sulle sinu töö eelmise versiooni taastamiseks pakub.

Salvestamise versioonid (Save Versions)

Kui sa faili käsitsi salvestad, käseb see säte Blenderit säilitada sinu praeguses töökataloogis vastavat arvu salvestatud faili versioone. Nende failide laiend on `.blend1`, `.blend2` ning see number kasvab kuni määratud versioonide hulga. Vanemad failid nimetatakse kõrgema numbriga. Näiteks vaikimisi sätte **2** puhul on sul failist kolm versiooni: `*.blend` (viimane salvestus), `*.blend1` (eelviimane salvestus) ja `*.blend2` (eel-eelviimane salvestus).

Salvesta ajutised failid automaatselt (Auto Save Temporary Files)

Sellesse kasti linnukese panemisel käseb Blenderil *automaatselt* salvestada käsilolevast tööst tagavara koopia, mis asub ajutiste failide kataloogis Temp (vaata selle asukohta kasutajaeelistuste akna User Preferences faili sakist File). See lülitab sisse ka valiku taimer (minutit) Timer(mins), mis määrab iga automaatse salvestuse vahelise aja minutites. Automaatselt salvestatud failid nimetatakse juhusliku numbriga ja neil on `.blend` laiend.

Automaatsete varukoopiate taastamine

Taasta automaatne varukoopia (Recover Auto Save)

Menüü File » Recover Auto Save... kaudu saad automaatselt salvestatud faili avada. Pärast sellise automaatse koopia avamist peaksid oma töökataloogis oleva parjasti kasutatava faili normaalse `.blend` failina üle salvestama.

Lae viimane sessioon (Recover Last Session)

Menüüvalik File » Recover Last Session avab `quit.blend` faili, mis salvestatakse Blenderi sulgemisel ajutisse kataloogi Temp. Pea meeles, et ajutises kataloogis Temp olevad failid kustutatakse arvuti taaskäivitamisel.

Tähtis märkus

Kui sa automaatse varukoopia avad, kaotad kõik muutused, mida peale viimast automaatset salvestamist tegid.

Iga projekti puhul eksisteerib ainult **üks** automaatne varukoopia (st Blender ei säilita neist varasemaid versioone — seega ei saa sa selle meetodi abil minna ajas tagasi rohkem kui paar minutit).

Teised sätted

Hiljutised failid Recent Files; See väli määrab, mitut viimati avatud faili näidatakse menüüs File » Open Recent.

Salvesta eelvaate pildid (Save Preview Images)

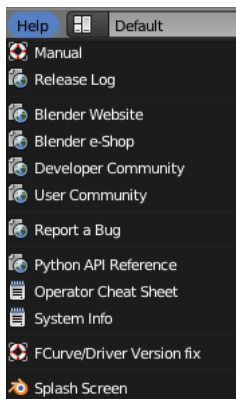
Failisirvija File Browser piltide ja materjalide eelvaated luuakse vastavalt vajadusele. Et neid eelvaateid oma `.blend` faili salvestada, lülita see valik sisse (selle tulemusena suureneb sinu `.blend` faili suurus).

Abisüsteem

Blenderil on valik erinevaid sisseehitatud ja veebipõhiseid abi võimalusi, mida saab kasutada infoakna Info päises oleva abimenüü Help kaudu.

Veebipääs

Mõned abivalikud käivitavad veebilehitseja ja suunavad selle Blenderi Sihtasutuse (*Blender Foundation*) serverisse. Selleks pead oma operatsioonisüsteemis määrama vaikesi kasutatava veebilehitseja. Kui sa kasutad sissehelistamisega internetiühendust, pead sa seadistama veebilehitseja looma ühenduse automaatselt, kui see pole hetkel aktiivne.



Abimenüü *Help*

Abimenüü (*Help*)

Veebipõhised abivõimalused

Manuaal (Manual)

Viib sind peamisele [wiki manuali](#) sisukorra lehele (inglise keeles).

Versioonilogi (Release Log)

Viib sind praeguse Blenderi versiooni [versioonimärkmete](#) veebilehele.

Blenderi kodulehekülg (Blender Website)

Viib sind [Blenderi](#) kodulehele.

Blenderi e-pood (Blender e-Shop)

Viib sind [Blenderi veebipoe](#) kodulehele.

Arenduskogukond (Developer Community)

Viib sind [blender.org kaasalõõmise](#) lehele. See leht käsitleb Blenderi tarkvaraarendust, vigade jahtimist (*Bug Tracking*), parandusi (*Patches*) ja skripte, haridust ja koolitust, dokumentatsiooni kirjutamist (*ja tõlkimist!* (*Tõlkija märkus*)) ning funktsionaalsuse uuringuid.

Kasutajate kogukond (User Community)

Viib sind lehele, kus on ära toodud nimerkiri Blenderi [kasutajate kogukonna veebilehtedest](#).

Teata veast (Report a Bug)

Viib sind [Blenderi veajahtija](#) lehele.

Pythoni API abi (Python API Reference)

Abilehed, mis kirjeldavad [Pythoni programmeerimisliidest \(API\)](#), mida Blender ja [Python](#) omavahelises suhtluses kasutavad.

Sisseehitatud abivõimalused

Operaatorite spikker (Operator Cheat Sheet)

Tekitab faili `OperatorList.txt`, kus on äratoodud kõik kasutatavad Pythoni operaatorid (*operators*). Seda saab lugeda tekstiredaktori kaudu. Vajuta võimalike tekstiplokkide nägemiseks menüü valikule Browse Text to be linked (see asub nupu New kõrval tekstiredaktori Text Editor akna päises).

Süsteemiinfo (System Info)

Süsteemiinfo loob tekstifaili `system-info.txt`, kus loetletakse sinu arvuti ja Blenderi olulised parameetrid, mis on kasulik vigade tuvastamisel. Seda nimekirja saab näha samal moel nagu ülaltoodud operaatorite spikri (Operator Cheat Sheet) omagi.


Draiverite parandus (FCurve/Driver fix)

Võtmevormide (*shapekey*) draiverite parandus.

Avaekraan (Splash Screen)

See näitab avaekraani, millelt saab teada programmi versiooni ja variandi..

Seadistusakna avamine

Seadistusakna (User preferences) avamiseks vali menüüst File » User Preferences või vajuta kiirklahvi CtrlAltU. Maci kasutajad peavad vajutama ⌘ Cmd,. Seadistusakna saad avada ka suvalises paanis, kui valid menüüst akna tüübiks  (seadistusaken User Preferences).



Selle redaktori abil saab häälestada Blenderi tööd. Valikud on jagatud 7-le kaardile (sakile), mille vahel saab valida akna ülaäärel asuvate järjekohjajate abil. Saksideks on: *Interface* (kasutajaliides), *Editing* (muutmine), *Input* (sisend), *Add-Ons* (lisamoodulid), *Themes* (teemad), *File* (failid) ja *System* (süsteem).

Seadistamine

Nüüd kui sa oled kasutajaeelistuste akna avanud, häälestame Blenderi just selliseks, nagu sa tahad! Vali järgnevast nimekirjast, mida sa tahad muuta:

[Kasutajaliides](#) • [Muutmine](#) • [Sisendid](#) • [Lisamoodulid](#) • [Teemad](#) • [Failieelistused](#) • [Süsteem](#)

Seadistuste salvestamine

Kui oled eelistusi muutnud, tuleb sul need salvestada. Vastasel juhul ei kasutata uusi seadeid Blenderi taaskäivitamisel ega isegi mitte järgmise stseeni loomisel. Blender salvestab eelistused koos iga stseeniga – see on mugav, kui sa vajad mingit kindlat ekraanipaigutust või lisamooduleid, kuid algaja võib sellest veidi ka segadusse sattuda.

Klõpsa kasutajaeelistuste akna päises (või antud puhul jaluses) nuppu Save As Default (salvesta vaikimisi seaded). See salvestab :

- Kõik uued eelistused
- Hetkel avatud stseeni vaikimisi stseenina.

Saad kasutajaeelistused salvestada ka kiirklahviga CtrlU.

Sissehitatud seadete taastamine

Vaikimisi Blenderi seadete taastamiseks algväärtustele on kaks võimalust:

1. Vali menüüst File » Load Factory Settings (fail -> laadi tehase seaded). Seejärel salvesta eelistused kiirvalikuga CtrlU või kasutajaeelistuste akna abil.
2. Kustuta oma arvutist fail `startup.blend`:
 - Linuxis asub see fail `/home/$user/.blender/'versiooni number'/config/startup.blend` (sa pead peidetud failide näitamise sisse lülitama)
 - Windows 7-s ja Windows Vistas, eeldusel et su operatsioonisüsteem on installeeritud C:\ kettale, on selle faili asukoht `C:\Users\%user%\AppData\Roaming\Blender Foundation\Blender\'versiooni number'\config\startup.blend`
 - MacOSis asub see fail `/Users/$user/Library/Application Support/Blender\'versiooni number'/config/startup.blend` (pead peidetud failide näitamise sisse lülitama)

Kui juba vaatad oma Blenderi konfiguratsioonikausta, tasub sinna salvestatud seaded ka kuskile mujale varuks kopeerida. Kui juhtumisi peaksid eelistused nii segamini ajama, et ei suuda neid enam tagasi muuta, saad sa alati kopeerida oma tagavaraks salvestatud seadete faili konfiguratsioonifaili asemele. Nii ennetad vajadust tehase seadete taastamiseks.

Ekraan (*Display*)

Tööriista vihjed (Tooltips)

Sisselülitamisel ilmub tööriista kohta vihje, kui hiire kursor selle kohal asetseb. See vihje seletab kursori alla jääva elemendi funktsioone ning toob ära ka sellega seotud kiirvaliku (kui see eksisteerib) ja sellele viitava Pyhtoni funktsiooni.

Objekti info (Object Info)

Näita 3D-vaate alumises vasakus nurgas aktiivse objekti nime ja valitud kaadri numbrit.

Suur kursor (Large Cursors)

Kasuta võimaluse korral suurt hiire kursorit.

Vaate nimi (View Name)

Näita 3D-akna ülemises vasakus nurga parajasti valitud vaate nime ja tüüpi. Näiteks: User Persp või Top Ortho.

Mahamängimise kaadrisagedus (Playback FPS)

Näita animatsiooni mängimise ajal ekraani värskenduse kaadrisagedust. See ilmub ühes vaateakna nurgas.

Globaalne stseen (Global Scene)

Parajasti valitud stseeni näidatakse ka kõigis teistes stseenides (projekt võib koosneda rohkem kui ühest stseenist).

Objekti keskme suurus (Object Origin Size)

3D-objektide keskkotade diameeter vaateaknas (väärtused pikslites 4 kuni 10).

Näita miniteljestikku (Display Mini Axis)

Näita vaateakna alumises vasakus nurgas miniteljestikku.

Suurus (Size)

Miniteljestiku suurus.

Heledus (Brightness)

Miniteljestiku heledus.

Omaduste aken (*Properties*)

Laiuse kontroll (Width Check)

Kui omaduste akna Properties window laius on sellest väärtusest väiksem, kuvatakse nuppe kahe tulba asemel ühes.

Vaate muutmine (*View Manipulation*)

Automaatne sügavus (Auto Depth)

Kasuta hiire alla jäävat sügavust, et parandada vaate liigutamist/pöörmist/suurendamist/vähendamist.

Suurenda hiire asukohas (Zoom to Mouse Position)

Sisselülitamisel kasutatakse suurendamise/vähendamise fookusena 2D-akna keskkoha asemel hiirekursori asukohta. See aitab sul vältida tasapinnalist liikumist, kui sa sageli vaadet suurendad või vähendad.

Pööra ümber valitu (Rotate Around Selection)

Valitud objekt muutub vaateakna pööramise keskpunktiks.

Globaalne keskpunkt (Global Pivot)

Lukusta kõigis 3D-vaadetes sama pööramise/möötkava teisendamise keskpunkt.

Automaatne perspektiiv (Auto Perspective)

Peale kasutajapoolt määratud paralleelprojektsiooni (User Orthographic) kasutamist lülitatakse ülalt/küljelt/eestvaatesse minnes projektsioon automaatselt perspektiivseks. Kui see on välja lülitatud, säilitatakse ülalt/küljelt/eestvaate muutmise puhul paralleelne või perspektiivne projektsioon (vastavalt sellele, kumb oli vaate muutmise ajal aktiivne).

Sujuv vaade (Smooth View)

Animatsiooni pikkus, mida näidatakse numbriklahvistikuga vaate muutmise puhul (ülalt/küljelt/eest/kaamera...). Animatsiooni eemaldamiseks vähendada nullini.

Pööramise nurk (Rotation Angle)

Pööramise sammude suurus kraadides, kui kasutada 3D-vaates pööramiseks klahve 4 NumPad, 6 NumPad, 8 NumPad või 2 NumPad.

2D-vaateaknad (*2D Viewports*)

Minimaalne ruudustiku vahe (Minimum Grid Spacing)

Minimaalne ruudustikujoonte vahe pikslites vaadatuna mõnes 2D-vaateaknas (nt ülalt paralleelne).

Ajakoodi stiil (TimeCode Style)

Ajakoodide formaat, kui neid ei näidata kaadrites. Formaat kasutab sekundist väiksemate kaadrite eraldamiseks märki '+' ning see võib olla vastavalt vajadusele kas vasakult või paremalt väiksemaks lõigatud.

Manipulaator (*Manipulator*)

Selle abil saad muuta objektide liigutamisel, pööramisel ja möötkava teisendamisel kasutatava 3D-teisendusmanipulaatori suurust (suurus (*Size*), sanga suurus (*Handle size*)).

Menüüd (*Menus*)

Ava, kui hiir on kohal (Open on Mouse Over)

Vali see, kui soovid, et Menüüd avaneksid nende kohal oleva hiire peale ning mitte nende vajutamisel.

Menüü avamise viivitus (Menu Open Delay)

Ülemisel tasemel (Top Level)

Ajaline viivitus 1/10 sekundites enne Menüü avanemist (hiire kohal olemise säte Open on Mouse Over peab olema sisse lülitatud).

Alamtasemel (Sub Level)

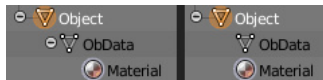
Sama, mis eelmine, aga alammenüüde puhul (näiteks: File » Open Recent).

Link Materials To (Lingi materjalid)



Valiku mõistmiseks pead teadma, kuidas Blenderi objektid töötavad. Peaaegu kõik Blenderis on paigutatud andmeblokkide hierarhjaatesse (Datablocks). Andmebloki võib pidada mahutiks, mis sisaldab teatud informatsiooni tükke. Näiteks objekti andmeblokk sisaldab andmeid selle asukoha kohta ja objektiandmete blokk (ObData) sisaldab infot võre enda kohta.

Materjal võib olla objektiga lingitud kahel viisil:



Objektiandmetega (ObData)
lingitud materjal (vasakul) ja
objektiga lingitud materjal
(paremal).

ObData (objektiandmed)

Iga loodud materjal luuakse osana objektiandmete andmeblokkist ObData (vasakul).

Object (objekt)

Iga loodud materjal luuakse osana objekti enda andmeblokkist (paremal).

Loe selle kohta lähemalt [siit](#).

New objects (uued objektid)

Enter Edit Mode (mine muutmisrežiimi)

Kui see valik on sisse lülitatud, aktiveeritakse uue objekti lisamisel automaatselt muutmisrežiim.

Align To (joonda)

World (maailm)

Uued objektid paigutatakse globaalse koordinaatsüsteemi järgi

View (vaade)

Uued objektid paigutatakse vaate koordinaatsüsteemi järgi

Undo (tühistamine)

Global Undo (globaalne tühistamine)

Hoiab mälu tervet dokumendi koopiat (see vajab rohkem mälu).

Step (samme)

Tühistamissammude maksimaalne arv.

Memory Limit (mälu piir)

Maksimaalne kasutatava mälu hulk megabaitides (0 tähendab piiramatult).

[Loe tühistamise ja uuesti tegemise kohta lähemalt »](#)

Grease Pencil (rasvakriit)

Rasvakriit võimaldab 3D-vaatesse "pliiatsiga" joonistada.

Manhattan Distance (Manhattani vahemaa)

Minimaalne pikslite arv, mille hiir peab kas vertikaal- või horisontaalsuunas liikuma, enne kui joont pikendatakse.

Euclidian Distance (Eukleidese vahemaa)

Minimaalne kaugus, mille hiir peab läbima, enne kui joont pikendatakse.

Eraser Radius (kustutaja raadius)

Koos rasvakriidiga kasutatava kustutaja raadius

Smooth Stroke (siledad jooned)

Määrab, kas valmis jooni silutakse

Playback (taasesitus)

Allow Negative Frame (luba negatiivseid kaadreid)

Kui see on sisse lülitatud, saab kasutada negatiivseid kaadriumbreid.

Keyframing (võtmekaadrid)

Paljudel juhtudel juhitakse animatsioone võtmekaadritega. Parameetri (näiteks asukoha) väärtus salvestatakse võtmekaadrisse ja kahe võtmekaadri vahelise animatsiooni leiab Blender interpoleerides.

Visual Keying (visuaalsed võtmed)

Kasuta piirajatega objektide jaoks automaatselt visuaalseid võtmeid.

Only Insert Needed (lisa ainult vajaduse korral)

Kui see on sisse lülitatud, lisatakse võtmekaader ainult siis, kui ta on vajalik.

Auto Keyframing (automaatsed võtmekaadrid)

Objektidele ja luudele lisatakse võtmekaadrid automaatselt. Automaatsed võtmekaadrid ei ole vaikimisi sisse lülitatud.

Only Insert Available (lisa ainult olemasolevatele)

Lisab automaatselt võtmekaadrid olemasolevatel kõveratel.

New F-Curve Defaults (uued F-kõverate vaikeseaded)

Interpolation (interpolatsioon)




See määrab, kuidas kahe võtmekaadri vaheline osa arvutatakse. Uute võtmekaadrite vaikimisi interpolatsioonirežiim: Bezier (sujuv kiirendamine ja aeglastumine), Linear (lineaarne) ja Constant (konstantne) (lülitub järsult ümber).

XYZ to RGB (XYZ RGB-ks)

X, Y ja Z animatsioonikõverate (asukoha, skaala või pöörde) värvid on needsamad, mis X, Y ja Z telgedel.

Transform (teisendus)

Release confirm (kinnita vabastamine)

objekti lohistamine LMB  hiirenupuga nihutab teda. Selle (ja teiste) teisenduste kinnitamiseks on vaikimisi vaja klõpata LMB . Kui see valik on sisse lülitatud, käitub LMB  vabastamine nagu teisenduse kinnitamine.

Duplicate Data (andmete duplitseerimine)

Seade 'Duplicate Data' (andmete duplitseerimine) määrab, millised andmed kopeeritakse objekti duplitseerimisel uuele objektile ja millised andmed lingitakse olemasolevatega. Kõigile linnukestega kastidele vastavad andmed kopeeritakse objekti duplitseerimisel täies mahus uuele objektile. Ilma linnukeseta kastidele vastavad andmed lingitakse duplitseeritud objektile esialgse objekti andmetest.

Näitena: kui sul on valik Mesh (võre) sisse lülitatud, siis iga võreobjekti duplikaadi jaoks luuakse uus täielik koopia võre geomeetriast ja iga võre käitub seega sõltumatult teistest võredest. Kui sa jäta valiku Mesh (võre) välja lülitatuks, siis ühe duplitseeritud objekti võre muutmine muudab täpselt samal viisil ka kõiki teisi kloone.

Samad reeglid kehtivad kõikide valikute puhul nimekirjast 'Duplicate Data' (andmete duplitseerimine).

Sisendi häälestamine

Blender võimaldab sul määrata mitu sisendite konfiguratsiooni häälestust (Preset). Vaikeklaviatuuritabeli kustutamise asemel võid lihtsalt ka uusi häälestusi luua. Nii saad edaspidi valida oma vaikeseaded vastavalt oma vajadustele. Ülaloleval pildil on hiire sätted leitavad akna vasakust servast ja klaviatuuri omad paremalt.

Häälestuste lisamine ja kustutamine



Enne vaikeseadetes millegi muutmist vajuta uue häälestuse (Preset) lisamiseks pildil näidatud *pluss* nuppu. Blender küsib sinult uue häälestuse nime, peale mida saad häälestuse nimekirjast muutmiseks valida. Kui soovid häälestust kustutada, vali see nimekirjast ja vajuta *miinus* nuppu.

Häälestuste valimine

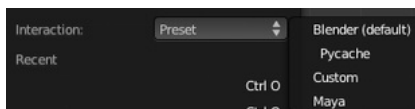
Saad oma parajasti kasutatavat häälestust muuta, kahel järgneval moel:

- Valides Blenderi käivitamisel avaekraanilt juhtimise menüü Interaction või kasutades menüüd Help » Splash Screen (abi -> avaekraan).
- Valides konfiguratsiooni kasutajaeelistuste aknast User Preferences Input.

Note

Pane tähele, et mõlemal juhul muudetakse eelhäälestust ainult hetkel avatud faili jaoks. Kui valid menüüst File » New (fail -> uus) või File » Open (fail -> ava), loetakse jälle sisse vaikehäälestus..

Vaikimisi häälestuse määramine



Kui sa oled oma hiire ja klaviatuuri häälestuse Presets konfigureerinud, saad määrata selle vaikimisi häälestuseks:

- Valides konfiguratsiooni häälestuse kasutajaeelistuste aknast User Preferences Input või
- Valides konfiguratsiooni häälestuse avaekraanilt.
- Määrates häälestuse vaikimisi häälestuseks kasutajaeelistuste aknas User Preferences nupuga Save As Default või klahvikombinatsiooniga CtrlU.

Klaviatuuri seadete eksport ja import

Mõnel juhul on sul vaja salvestada klaviatuuri seaded välisesse faili (nt kui tahad uut süsteemi installeerida või oma seadeid teiste kasutajatega jagada...). Vajuta nuppu Export Key Configuration (eksporti klaviatuuriseaded) kaardil Input (sisend) ja avaneb failisirvija, milles saab määrata, kuhu seadistused salvestada. Nupp Import Key Configuration (impordi klaviatuuriseaded) installeerib klaviatuuri seadistused, mis on failina sinu arvutis, kuid mitte Blenderis.

Mouse (hiir)

Emulate 3 Button Mouse (emuleeri 3 nupuga hiirt)

Blenderit on võimalik kasutada ilma 3-nupulise hiireta (näiteks kahenupulise hiirega või Apple'i ühenupulise hiirega või sülearvutil). Seda funktsionaalsust saab emuleerida klahvide ja hiire kombinatsiooniga. See valik on lubatud ainult siis, kui välja Select With (vali) väärtuseks on määratud Right (parem). [Loe lähemalt 3 nupuga hiire emuleerimise kohta »](#)

Continuous Grab (pidev haare)

Lubab hiirel liikuda väljapoole akent (näiteks nihutamiseks, pööramiseks ja skaleerimiseks).

Drag Threshold (lohistamise lävi)

Pikslite arv, mille võrra kasutajaliidese elementi tuleb enne liigutada, kui Blender seda liigutamiseks peab.

Select with (vali)

Saad määrata, millist hiirenuppu kasutatakse valimiseks (teist kasutatakse 3D-kursori paigutamiseks).

Double Click (topeltklõps)


Topeltklõpsu viiteaeg (millisekundites).

Numbrivälja emulatsioon

Klaviatuuri numbrivälja klahve kasutatakse Blenderis üpris sageli ja nad ei käitu samamoodi nagu tavalised numbriklahvid. Kui sul on ilma numbriväljata klaviatuur (näiteks sülearvutil), saad sa Blenderis nende asemel kasutada tavalisi numbriklahve. Lihtsalt lülita valik Emulate Numpad (emuleeri numbrivälja) sisse.

Vaate muutmine


Orbit Style (tiirlemisviis)

Määrab, kuidas Blender käitub, kui sa pöörad 3D-vaadet (vaikimisi MMB ). Võimalikud on kaks lahendust. Kui kasutasid varem programme Maya või Cinema 4D, siis eelistad ilmselt varianti Turntable (plaadimängija).


Zoom Style (suurendusviis)

Vali eelistatud viis suurendamiseks ja vähendamiseks Ctrl MMB 

Scale (skaala)

Suurendus sõltub sellest, kus vaate piirkonnas sa esmalt klõpsad. Kombinatsiooniga Ctrl MMB  hiirt sellest punktist akna keskme suunas liigutades suurendus väheneb, akna serva suunas liigutades suurendus kasvab.

Continue (jätka)

Selle stiili puhul juhid sa suurendamise kiirust (mitte määra), liigutades hiirt esmase Ctrl MMB  vajutamise kohast eemale. Hiirt sellest kohast üles või paremale liigutades suurendus väheneb, alla või vasakule liigutades kasvab. Mida kaugemale hiirt liigutad, seda kiiremini suurendus muutub. Suunda saab muuta valiknuppude Vertical (vertikaalne) ja Horizontal (horisontaalne) abil ning Invert Zoom Direction (muuda suurenduse suunda) sätte abil.

Dolly (käru)

See valik töötab sarnaselt jätka valikuga Continue, kuid suurendamise kiirus on konstantne.

Vertical (vertikaalne)

Hiire liigutamine ülespoole vähendab suurendust, allapoole suurendab.

Horizontal (horisontaalne)

Paremale liigutamine vähendab suurendust, vasakule suurendab.

Invert Zoom Direction (muuda hiire suunda)

Muudab käru Dolly ja jätka Continue suurendamise puhul liigutamise suunad.

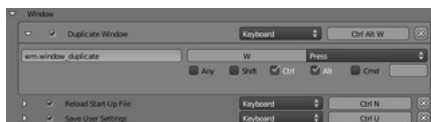
Invert Wheel Zoom Direction (muuda hiireratta suunda)

Muudab hiirerattaga suurendamise suunda.

NDOF device (3D-hiir)

Määra 3D-hiire tundlikus.

Kiirklahvide redaktor



Kiirklahvide redaktori abil saad sa muuta vaikimisi määratud kiirklahve. Sa saad muuta iga akna kiirklahve eraldi.

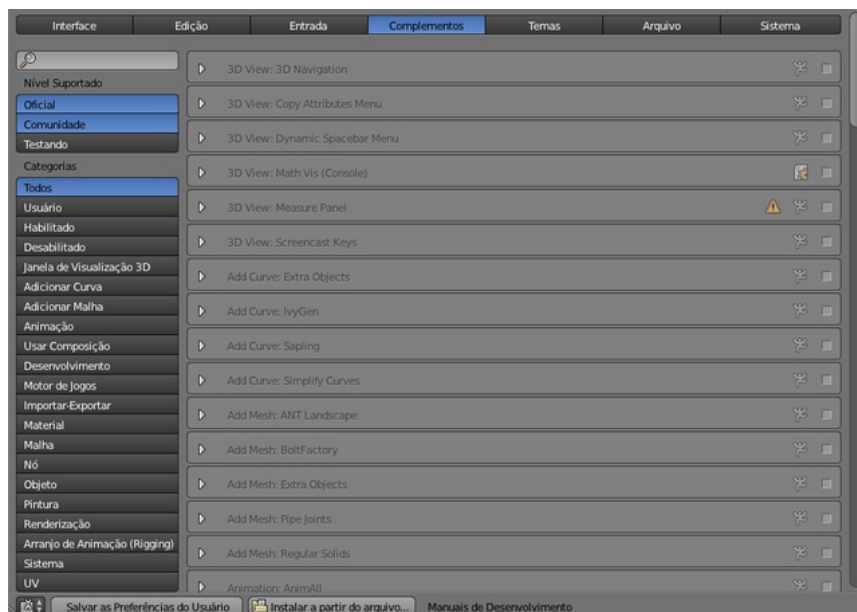
1. Vali kiirklahv, mida tahad muuta, ning vajuta kiirklahvitabeli avamiseks valgetele nooltele.
2. Määra, milline sisend teostab vastava protseduuri
 - Keyboard (klaviatuur): Ainult kiirklahv või modifitseerijatega kiirklahv (E või ⇧ ShiftE).
 - Mouse (hiir): Vasak/keskmise/parem hiirenupp. Seda saab kombineerida modifikaatorklahvidega Alt, ⇧ Shift, Ctrl, ⌘ Cmd.
 - Tweak (sikuta): Klõpsa ja lohista. Saab samuti kombineerida eelpooltoodud klahvidega.
 - Text input (tekstisisestus): Protseduuri käivitab teksti sisestamine
 - Timer (taimer): Kasutatakse ajavahemikust sõltuvate tegevuste määramiseks (nt animatsiooni samm (Animation Step) kasutab vaikimisi taimerit väärtusega 0, vaate siluja (Smooth view) kasutab taimerit väärtusega 1).
3. Muuda kiirklahve nii nagu soovid. Lihtsalt klõpsa kiirklahvi sisendi väljal ja sisesta uus kiirklahv.

Kui tahad taastada klaviatuuritabeli vaikeväärtusi, vajuta nuppu Restore (taasta) tabeli paremas ülanurgas .

[Kasutajaliides](#) • [Muutmine](#) • [Sisendid](#) • **Lisamoodulid** • [Teemad](#) • [Failieelistused](#) • [Süsteem](#)

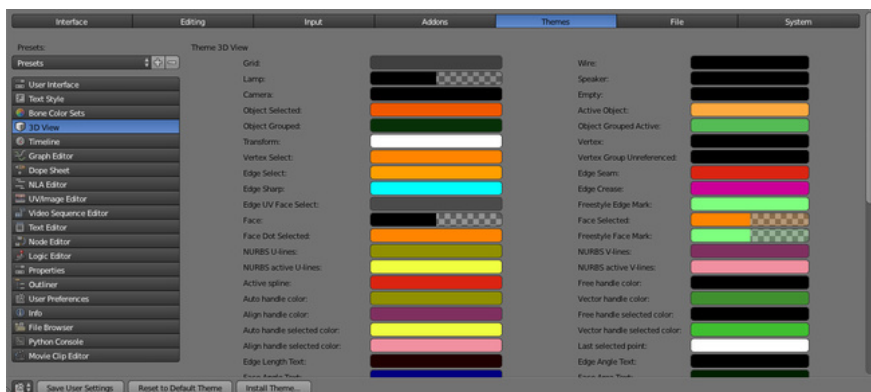
Lisamoodulite (*Add-Ons*) kaardil saab hallata täiendavaid protseduure, mis ei ole Blenderis vaikimisi sisse lülitatud. Uusi funktsioone saab lisada nupuga *Install Add-Ons* (lisa mooduleid). Blenderi kogukonna loodud lisamooduliste arv kasvab pidevalt — kui otsid mõnd vajalikku meetodit või protseduuri, vaata ringi (või kirjuta see ise).

Vaata lisamoodulite kasutamise kohta lähemalt [lisamoodulite lehelt](#).



Teemade seadistamine

Nagu juba varem öeldud, on Blenderi kasutajaliideses võimalik väga paljut muuta—osad seadistused muudavad kasutajaliidese väljanägemist ja värve. Need asuvad kaardil Themes (teemad).



Iga redaktori värve saab eraldi määrata—lihtsalt vali menüüst soovitud redaktor ja muuda värve, nagu tahad. Pane tähele, et muudatused on ekraanil kohe näha. Lisaks on võimalik muuta selliseid detaile nagu 3D-vaate või graafikuredaktori punktide suurus.

Teemade muutmine Pythoni abil

Kui tahad muuta teemat skriptiga, siis siin on näidisskript, kus parameetrid on ükshaaval välja toodud:

```
# Lihtne viis muuta globaalse teema värve.
```

```
import bpy
```

```
#Jagatud elemendid
back = (0.2,0.2,0.2)
back_alt = (0.15,0.15,0.15)
button = (0.2,0.2,0.2)
grid = (0.3,0.3,0.3)
header = (0.2,0.2,0.2)
panel = (0.2,0.2,0.2,0.75)
text = (0.8,0.8,0.8)
text_hi = (1,1,1)
wire = (1,1,1)
lamp = (1,1,1,1)
```

```
#Kasutajaliidesel jagatud
inner = (0.2,0.2,0.2,1)
inner_sel = (0.2,0.2,0.2,1)
item = (0.2,0.2,0.2,1)
outline = (0.1,0.1,0.1)
```

```
theme = bpy.context.user_preferences.themes['Default']
```

```
theme.view_3d.act_spline
theme.view_3d.back = back_alt
theme.view_3d.bone_pose
theme.view_3d.bone_solid
theme.view_3d.button= button
theme.view_3d.button_text = text
theme.view_3d.button_text_hi = text_hi
theme.view_3d.button_title = text
theme.view_3d.edge_crease
theme.view_3d.edge_facesel
theme.view_3d.edge_seam
theme.view_3d.edge_select
theme.view_3d.edge_sharp
theme.view_3d.editmash_active
theme.view_3d.face
theme.view_3d.face_dot
theme.view_3d.face_select
theme.view_3d.facedot_size
theme.view_3d.frame_current
theme.view_3d.grid = grid
theme.view_3d.handle_align
theme.view_3d.handle_auto
theme.view_3d.handle_free
theme.view_3d.handle_sel_align
theme.view_3d.handle_sel_auto
theme.view_3d.handle_sel_free
theme.view_3d.handle_sel_vect
theme.view_3d.handle_vect
theme.view_3d.header = header
theme.view_3d.header_text = text
theme.view_3d.header_text_hi = text_hi
theme.view_3d.lamp = lamp
theme.view_3d.lastsel_point
theme.view_3d.normal
theme.view_3d.nurb_sel_uline
theme.view_3d.nurb_sel_vline
theme.view_3d.nurb_uline
theme.view_3d.nurb_vline
theme.view_3d.object_active
theme.view_3d.object_grouped
```

```
theme.view_3d.object_grouped = active
theme.view_3d.object_selected
theme.view_3d.panel = panel
theme.view_3d.text = text
theme.view_3d.text_hi = text_hi
theme.view_3d.title = text
theme.view_3d.transform
theme.view_3d.vertex
theme.view_3d.vertex_normal
theme.view_3d.vertex_select
theme.view_3d.vertex_size
theme.view_3d.wire = wire
```

```
theme.console.back= back_alt
theme.console.button = button
theme.console.button_text = text
theme.console.button_text_hi = text_hi
theme.console.button_title = text
theme.console.cursor
theme.console.header = header
theme.console.header_text = text
theme.console.header_text_hi = text_hi
theme.console.line_error
theme.console.line_info
theme.console.line_input
theme.console.line_output
theme.console.text = text
theme.console.text_hi = text_hi
theme.console.title = text
```

```
theme.dopesheet_editor.active_channels_group
theme.dopesheet_editor.back= back
theme.dopesheet_editor.button = button
theme.dopesheet_editor.button_text = text
theme.dopesheet_editor.button_text_hi = text_hi
theme.dopesheet_editor.button_title = text
theme.dopesheet_editor.channel_group
theme.dopesheet_editor.channels
theme.dopesheet_editor.channels_selected
theme.dopesheet_editor.dopesheet_channel
theme.dopesheet_editor.dopesheet_subchannel
theme.dopesheet_editor.frame_current
theme.dopesheet_editor.grid = grid
theme.dopesheet_editor.header = header
theme.dopesheet_editor.header_text = text
theme.dopesheet_editor.header_text_hi = text_hi
theme.dopesheet_editor.list = button
theme.dopesheet_editor.list_text = text
theme.dopesheet_editor.list_text_hi = text
theme.dopesheet_editor.list_title = text
theme.dopesheet_editor.long_key
theme.dopesheet_editor.long_key_selected
theme.dopesheet_editor.text = text
theme.dopesheet_editor.text_hi = text_hi
theme.dopesheet_editor.title = text
theme.dopesheet_editor.value_sliders
theme.dopesheet_editor.view_sliders
```

```
theme.file_browser.active_file
theme.file_browser.active_file_text
theme.file_browser.back= back
theme.file_browser.button = button
theme.file_browser.button_text = text
theme.file_browser.button_text_hi = text_hi
theme.file_browser.button_title = text
theme.file_browser.header = header
theme.file_browser.header_text = text
theme.file_browser.header_text_hi = text_hi
theme.file_browser.list = button
theme.file_browser.list_text = text
theme.file_browser.list_text_hi = text
theme.file_browser.list_title = text
theme.file_browser.scroll_handle = button
theme.file_browser.scrollbar = button
theme.file_browser.selected_file
theme.file_browser.text = text
theme.file_browser.text_hi = text_hi
theme.file_browser.tiles = back
theme.file_browser.title = text
```

```
theme.graph_editor.active_channels_group
theme.graph_editor.back= back_alt
theme.graph_editor.button = button
theme.graph_editor.button_text = text
theme.graph_editor.button_text_hi = text_hi
theme.graph_editor.button_title = text
theme.graph_editor.channel_group
theme.graph_editor.channels_region
theme.graph_editor.dopesheet_channel
theme.graph_editor.dopesheet_subchannel
theme.graph_editor.frame_current
theme.graph_editor.grid = grid
theme.graph_editor.handle_align
theme.graph_editor.handle_auto
```

```
theme.graph_editor.handle_free
theme.graph_editor.handle_sel_align
theme.graph_editor.handle_sel_auto
theme.graph_editor.handle_sel_free
theme.graph_editor.handle_sel_vect
theme.graph_editor.handle_vect
theme.graph_editor.handle_vertex
theme.graph_editor.handle_vertex_select
theme.graph_editor.handle_vertex_size
theme.graph_editor.header = header
theme.graph_editor.header_text = text
theme.graph_editor.header_text_hi = text_hi
theme.graph_editor.lastsel_point
theme.graph_editor.list = button
theme.graph_editor.list_text = text
theme.graph_editor.list_text_hi = text
theme.graph_editor.list_title = text
theme.graph_editor.panel = back
theme.graph_editor.text = text
theme.graph_editor.text_hi = text_hi
theme.graph_editor.title = text
theme.graph_editor.vertex
theme.graph_editor.vertex_select
theme.graph_editor.vertex_size
theme.graph_editor.window_sliders
```

```
theme.image_editor.back= back_alt
theme.image_editor.button = button
theme.image_editor.button_text = text
theme.image_editor.button_text_hi = text_hi
theme.image_editor.button_title = text
theme.image_editor.editmesh_active
theme.image_editor.face
theme.image_editor.face_dot
theme.image_editor.face_select
theme.image_editor.facedot_size
theme.image_editor.header = header
theme.image_editor.header_text = text
theme.image_editor.header_text_hi = text_hi
theme.image_editor.scope_back
theme.image_editor.text = text
theme.image_editor.text_hi = text_hi
theme.image_editor.title = text
theme.image_editor.vertex
theme.image_editor.vertex_select
theme.image_editor.vertex_size
```

```
theme.info.back= back
theme.info.button = button
theme.info.button_text = text
theme.info.button_text_hi = text_hi
theme.info.button_title = text
theme.info.header = header
theme.info.header_text = text
theme.info.header_text_hi = text_hi
theme.info.text = text
theme.info.text_hi = text_hi
theme.info.title = text
```

```
theme.logic_editor.back= back
theme.logic_editor.button = button
theme.logic_editor.button_text = text
theme.logic_editor.button_text_hi = text_hi
theme.logic_editor.button_title = text
theme.logic_editor.header = header
theme.logic_editor.header_text = text
theme.logic_editor.header_text_hi = text_hi
theme.logic_editor.panel = back
theme.logic_editor.text = text
theme.logic_editor.text_hi = text_hi
theme.logic_editor.title = text
```

```
theme.nla_editor.back= back
theme.nla_editor.bars
theme.nla_editor.bars_selected
theme.nla_editor.button = button
theme.nla_editor.button_text = text
theme.nla_editor.button_text_hi = text_hi
theme.nla_editor.button_title = text
theme.nla_editor.frame_current
theme.nla_editor.grid = grid
theme.nla_editor.header = header
theme.nla_editor.header_text = text
theme.nla_editor.header_text_hi = text_hi
theme.nla_editor.list = button
theme.nla_editor.list_text = text
theme.nla_editor.list_text_hi = text
theme.nla_editor.list_title = text
theme.nla_editor.strips
theme.nla_editor.strips_selected
theme.nla_editor.text = text
theme.nla_editor.text_hi = text_hi
theme.nla_editor.title = text
```

```
theme.nla_editor.view_sliders
```

```
theme.node_editor.back= back_alt
theme.node_editor.button = button
theme.node_editor.button_text = text
theme.node_editor.button_text_hi = text_hi
theme.node_editor.button_title = text
theme.node_editor.converter_node
theme.node_editor.group_node
theme.node_editor.header = header
theme.node_editor.header_text = text
theme.node_editor.header_text_hi = text_hi
theme.node_editor.in_out_node
theme.node_editor.list = button
theme.node_editor.list_text = text
theme.node_editor.list_text_hi = text
theme.node_editor.list_title = text
theme.node_editor.node_backdrop
theme.node_editor.operator_node
theme.node_editor.selected_text
theme.node_editor.text = text
theme.node_editor.text_hi = text_hi
theme.node_editor.title = text
theme.node_editor.wire
theme.node_editor.wire_select
```

```
theme.outliner.back= back
theme.outliner.button = button
theme.outliner.button_text = text
theme.outliner.button_text_hi = text_hi
theme.outliner.button_title = text
theme.outliner.header = header
theme.outliner.header_text = text
theme.outliner.header_text_hi = text_hi
theme.outliner.text = text
theme.outliner.text_hi = text_hi
theme.outliner.title = text
```

```
theme.properties.back= back
theme.properties.button = button
theme.properties.button_text = text
theme.properties.button_text_hi = text_hi
theme.properties.button_title = text
theme.properties.header = header
theme.properties.header_text = text
theme.properties.header_text_hi = text_hi
theme.properties.panel = back
theme.properties.text = text
theme.properties.text_hi = text_hi
theme.properties.title = text
```

```
theme.sequence_editor.audio_strip
theme.sequence_editor.back= back
theme.sequence_editor.button = button
theme.sequence_editor.button_text = text
theme.sequence_editor.button_text_hi = text_hi
theme.sequence_editor.button_title = text
theme.sequence_editor.draw_action
theme.sequence_editor.effect_strip
theme.sequence_editor.frame_current
theme.sequence_editor.grid = grid
theme.sequence_editor.header = header
theme.sequence_editor.header_text = text
theme.sequence_editor.header_text_hi = text_hi
theme.sequence_editor.image_strip
theme.sequence_editor.keyframe
theme.sequence_editor.meta_strip
theme.sequence_editor.movie_strip
theme.sequence_editor.plugin_strip
theme.sequence_editor.scene_strip
theme.sequence_editor.text = text
theme.sequence_editor.text_hi = text_hi
theme.sequence_editor.title = text
theme.sequence_editor.transition_strip
theme.sequence_editor.window_sliders
```

```
theme.text_editor.back= back_alt
theme.text_editor.button = button
theme.text_editor.button_text = text
theme.text_editor.button_text_hi = text_hi
theme.text_editor.button_title = text
theme.text_editor.cursor
theme.text_editor.header = header
theme.text_editor.header_text = text
theme.text_editor.header_text_hi = text_hi
theme.text_editor.line_numbers_background
theme.text_editor.scroll_bar = back
theme.text_editor.selected_text
theme.text_editor.syntax_builtin
theme.text_editor.syntax_comment
theme.text_editor.syntax_numbers
theme.text_editor.syntax_special
```

```
theme.text_editor.syntax_string
theme.text_editor.text = text
theme.text_editor.text_hi = text_hi
theme.text_editor.title = text
```

```
theme.timeline.back= back_alt
theme.timeline.button = button
theme.timeline.button_text = text
theme.timeline.button_text_hi = text_hi
theme.timeline.button_title = text
theme.timeline.frame_current
theme.timeline.grid = grid
theme.timeline.header = header
theme.timeline.header_text = text
theme.timeline.header_text_hi = text_hi
theme.timeline.text = text
theme.timeline.text_hi = text_hi
theme.timeline.title = text
```

```
theme.user_preferences.back= back
theme.user_preferences.button = button
theme.user_preferences.button_text = text
theme.user_preferences.button_text_hi = text_hi
theme.user_preferences.button_title = text
theme.user_preferences.header = header
theme.user_preferences.header_text = text
theme.user_preferences.header_text_hi = text_hi
theme.user_preferences.text = text
theme.user_preferences.text_hi = text_hi
theme.user_preferences.title = text
```

```
theme.user_interface.icon_file
```

```
theme.user_interface.wcol_box.inner = inner
theme.user_interface.wcol_box.inner_sel
theme.user_interface.wcol_box.item = item
theme.user_interface.wcol_box.outline = outline
theme.user_interface.wcol_box.shadedown
theme.user_interface.wcol_box.shadetop
theme.user_interface.wcol_box.show_shaded
theme.user_interface.wcol_box.text = text
theme.user_interface.wcol_box.text_sel = text_hi
```

```
theme.user_interface.wcol_list_item.inner = inner
theme.user_interface.wcol_list_item.inner_sel
theme.user_interface.wcol_list_item.item = item
theme.user_interface.wcol_list_item.outline = outline
theme.user_interface.wcol_list_item.shadedown
theme.user_interface.wcol_list_item.shadetop
theme.user_interface.wcol_list_item.show_shaded
theme.user_interface.wcol_list_item.text = text
theme.user_interface.wcol_list_item.text_sel = text_hi
```

```
theme.user_interface.wcol_menu.inner = inner
theme.user_interface.wcol_menu.inner_sel
theme.user_interface.wcol_menu.item = item
theme.user_interface.wcol_menu.outline = outline
theme.user_interface.wcol_menu.shadedown
theme.user_interface.wcol_menu.shadetop
theme.user_interface.wcol_menu.show_shaded
theme.user_interface.wcol_menu.text = text
theme.user_interface.wcol_menu.text_sel = text_hi
```

```
theme.user_interface.wcol_menu_back.inner = inner
theme.user_interface.wcol_menu_back.inner_sel
theme.user_interface.wcol_menu_back.item = item
theme.user_interface.wcol_menu_back.outline = outline
theme.user_interface.wcol_menu_back.shadedown
theme.user_interface.wcol_menu_back.shadetop
theme.user_interface.wcol_menu_back.show_shaded
theme.user_interface.wcol_menu_back.text = text
theme.user_interface.wcol_menu_back.text_sel = text_hi
```

```
theme.user_interface.wcol_menu_item.inner
theme.user_interface.wcol_menu_item.inner_sel
theme.user_interface.wcol_menu_item.item = item
theme.user_interface.wcol_menu_item.outline = outline
theme.user_interface.wcol_menu_item.shadedown
theme.user_interface.wcol_menu_item.shadetop
theme.user_interface.wcol_menu_item.show_shaded
theme.user_interface.wcol_menu_item.text = text
theme.user_interface.wcol_menu_item.text_sel = text_hi
```

```
theme.user_interface.wcol_num.inner = inner
theme.user_interface.wcol_num.inner_sel
theme.user_interface.wcol_num.item = item
theme.user_interface.wcol_num.outline = outline
theme.user_interface.wcol_num.shadedown
theme.user_interface.wcol_num.shadetop
theme.user_interface.wcol_num.show_shaded
theme.user_interface.wcol_num.text = text
theme.user_interface.wcol_num.text_sel = text_hi
```

```
theme.user_interface.wcol_numslider.inner = inner
```



```
theme.user_interface.wcol_numslider.inner_sel
theme.user_interface.wcol_numslider.item = item
theme.user_interface.wcol_numslider.outline = outline
theme.user_interface.wcol_numslider.shadedown
theme.user_interface.wcol_numslider.shadetop
theme.user_interface.wcol_numslider.show_shaded
theme.user_interface.wcol_numslider.text = text
theme.user_interface.wcol_numslider.text_sel = text_hi
```

```
theme.user_interface.wcol_option.inner = inner
theme.user_interface.wcol_option.inner_sel
theme.user_interface.wcol_option.item = item
theme.user_interface.wcol_option.outline = outline
theme.user_interface.wcol_option.shadedown
theme.user_interface.wcol_option.shadetop
theme.user_interface.wcol_option.show_shaded
theme.user_interface.wcol_option.text = text
theme.user_interface.wcol_option.text_sel = text_hi
```

```
theme.user_interface.wcol_progress.inner = inner
theme.user_interface.wcol_progress.inner_sel
theme.user_interface.wcol_progress.item = item
theme.user_interface.wcol_progress.outline = outline
theme.user_interface.wcol_progress.shadedown
theme.user_interface.wcol_progress.shadetop
theme.user_interface.wcol_progress.show_shaded
theme.user_interface.wcol_progress.text = text
theme.user_interface.wcol_progress.text_sel = text_hi
```

```
theme.user_interface.wcol_pulldown.inner = inner
theme.user_interface.wcol_pulldown.inner_sel
theme.user_interface.wcol_pulldown.item = item
theme.user_interface.wcol_pulldown.outline = outline
theme.user_interface.wcol_pulldown.shadedown
theme.user_interface.wcol_pulldown.shadetop
theme.user_interface.wcol_pulldown.show_shaded
theme.user_interface.wcol_pulldown.text = text
theme.user_interface.wcol_pulldown.text_sel = text_hi
```

```
theme.user_interface.wcol_radio.inner = inner
theme.user_interface.wcol_radio.inner_sel
theme.user_interface.wcol_radio.item = item
theme.user_interface.wcol_radio.outline = outline
theme.user_interface.wcol_radio.shadedown
theme.user_interface.wcol_radio.shadetop
theme.user_interface.wcol_radio.show_shaded
theme.user_interface.wcol_radio.text = text
theme.user_interface.wcol_radio.text_sel = text_hi
```

```
theme.user_interface.wcol_regular.inner = inner
theme.user_interface.wcol_regular.inner_sel
theme.user_interface.wcol_regular.item = item
theme.user_interface.wcol_regular.outline = outline
theme.user_interface.wcol_regular.shadedown
theme.user_interface.wcol_regular.shadetop
theme.user_interface.wcol_regular.show_shaded
theme.user_interface.wcol_regular.text = text
theme.user_interface.wcol_regular.text_sel = text_hi
```

```
theme.user_interface.wcol_scroll.inner = inner
theme.user_interface.wcol_scroll.inner_sel
theme.user_interface.wcol_scroll.item = item = inner
theme.user_interface.wcol_scroll.outline = outline
theme.user_interface.wcol_scroll.shadedown
theme.user_interface.wcol_scroll.shadetop
theme.user_interface.wcol_scroll.show_shaded
theme.user_interface.wcol_scroll.text = text
theme.user_interface.wcol_scroll.text_sel = text_hi
```

```
theme.user_interface.wcol_state.inner_anim
theme.user_interface.wcol_state.inner_anim_sel
theme.user_interface.wcol_state.inner_driven
theme.user_interface.wcol_state.inner_driven_sel
theme.user_interface.wcol_state.inner_key
theme.user_interface.wcol_state.inner_key_sel
```

```
theme.user_interface.wcol_text.inner = inner
theme.user_interface.wcol_text.inner_sel
theme.user_interface.wcol_text.item = item
theme.user_interface.wcol_text.outline = outline
theme.user_interface.wcol_text.shadedown
theme.user_interface.wcol_text.shadetop
theme.user_interface.wcol_text.show_shaded
theme.user_interface.wcol_text.text = text
theme.user_interface.wcol_text.text_sel = text_hi
```

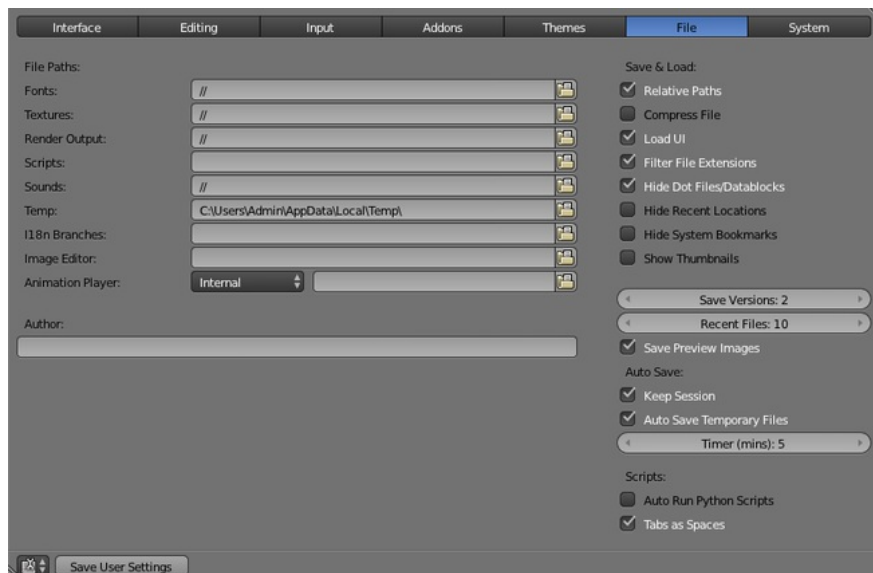
```
theme.user_interface.wcol_toggle.inner = inner
theme.user_interface.wcol_toggle.inner_sel
theme.user_interface.wcol_toggle.item = item
theme.user_interface.wcol_toggle.outline = outline
theme.user_interface.wcol_toggle.shadedown
theme.user_interface.wcol_toggle.shadetop
theme.user_interface.wcol_toggle.show_shaded
theme.user_interface.wcol_toggle.text = text
theme.user_interface.wcol_toggle.text_sel = text_hi
```

```
theme.user_interface.wcol_tool.inner = inner
```

```
theme.user_interface.wcol_tool.inner_sel
theme.user_interface.wcol_tool.item = item
theme.user_interface.wcol_tool.outline = outline
theme.user_interface.wcol_tool.shadedown
theme.user_interface.wcol_tool.shadetop
theme.user_interface.wcol_tool.show_shaded
theme.user_interface.wcol_tool.text = text
theme.user_interface.wcol_tool.text_sel = text_hi
```

Failieelistused

Allpool kirjeldame pildil nähtavaid failieelistusi.



File Paths (failide asukohad)

Kui sa töötad tähtsa projektiga, on mõistlik failide asukohad seadistada. Määra erinevatele kasutatavatele failitüüpidele vaikumisi kasutatavad kataloogid.

Siin on näide konfiguratsioonist.

Fonts (kirjatüübid)	//fonts/
Textures (tekstuurid)	//textures/
Texture Plugins (tekstuuripluginad)	//plugins/texture/
Sequence Plugins (montaažipluginad)	//plugins/sequence/
Render Output (renderduse väljund)	//renders/
Scripts (skriptid)	//scripts/
Sounds (helid)	//sounds/
Temp (ajutine)	//tmp/

Pane tähele, et Blender ei loo automaatselt projekti kataloogipuud. Sa pead kõik kataloogid failisirvijas käsitsi looma.

Save & Load (salvestamine ja lugemine)

Relative Paths (suhtelised asukohad)

Vaikumisi loetakse välised andmed kataloogist, mis arvestab .blend-faili asukohaga. See töötab muidugi ainult siis, kui Blenderi fail on juba salvestatud.

Compress File (paki fail)

Pakib .blend-faili salvestamisel kokku

Load UI (loe kasutajaliides)

Vaikumisi loetakse .blend-faili lugemisel ka sinna salvestatud kasutajaliidese seadistus (vt [ekraanipaigutus](#)). Seda saab aga iga faili lugemisel eraldi määrata kasutades failisirvija akna (File Browser) Blenderi failide avamise paneeli (Open Blender File).



Faililaiendi filter

Filter File Extensions (filtreeri faililaiendid)

Kui see on sisse lülitatud, näitab failisirvija ainult õiget tüüpi faile (st Blenderi dokumendi avamisel ainult .blend-faile).

Failitüüpide valikut saab muuta failisirvija aknas.

Hide Dot File/Datablocks (peida punktidega failid/andmehlokid)

Peidab failisirvijas failid, mille alguses on "." (Linux ja Apple operatsioonisüsteemides on "." failid peidetud).

Hide Recent Locations (peida hiljutine asukoht)

Peidab failisirvija hiljuti kasutatud katalooge näitava paneeli Recent.

Show Thumbnails (näita eelvaateid)

Näitab failisirvijas piltide ja videode puhul eelvaateid.

Auto Save (automaatsalvestus)

Save Versions (salvesta versioone)

Samast failist loodavate versioonide (varukoopiate) arv.

Recent Files (hiljutised failid)

Failide arv, mida näidatakse menüüs File » Open Recent (fail -> ava hiljutine).

Save Preview Images (salvesta eelvaade)

Piltide ja materjalide eelvaated luuakse failisirvijas automaatselt. Kui soovid need eelvaated oma .blend-faili salvestada, lülita see valik sisse (sellisel juhul su faili suurus kasvab).

Auto Save Temporary File (salvesta automaatselt ajutine fail)

Lülitab sisse automaatse salvestuse (loob ajutise faili)

Timer (taimer)

Aeg automaatsete salvestuste vahel.

[Loe automaatse salvestamise sätete kohta edasi »](#)

Süsteemieelistused (*System preferences*)

Pildil on kujutatud Blenderi kasutajaeelistuste akna User Preferences süsteemieelistuste saki System. Selle seadistusi selgitatakse allpool.



General (üldine)

- DPI
 - Määrab ekraani lahutuse ja seega ka kasutajaliidese fontide suuruse. Et muuta ainult kasutajaliidese elementide suurust paneelil, on sul tõenäoliselt targem hoida all klahvi Ctrl ning lohistada MMB abil hiirt paremale ning vasakule.
- Frame Server Port (kaadriserveri port)
 - Kaadriserveri pordi number renderdamiseks. Kasutatakse jagatud renderdamise puhul.
- Console Scrollback (konsooli tagasikerimine)
 - Tekstiridade arv, mida konsool mälus hoiab.
- Auto Run Python Scripts (käivita Pythoni skriptid automaatselt)
 - Lubab `.blend`-failidel käivitada neis leiduvad skriptid automaatselt (see võib olla üliohtlik, kui failid pärinevad kahtlasest allikast).
- Tabs as Spaces (tabulaatorid tühikuteks)
 - Teksti kirjutamise ja failide lugemise ajal asendatakse tabulaatorid tekstifailides tühikutega.

Sound (heli)

- Sound (heli)
 - Määra heli väljundseade.
- Channels (kanalid)
 - Määra helikanalite arv.
- Mixing Buffer (miksispühver)
 - Määra mitut sãmplit heli miksimispühver kasutab.
- Sample Rate (sãmplisagedus)
 - Määra heli sãmplisagedus.
- Sample Format (sãmpli formaat)
 - Määra heli sãmpli formaat.

Screenshot (ekraanivideo)

- FPS
 - Kaadrisagedus ekraanivideo mahamãngimisel
- Wait Timer (taimer)
 - Aeg millisekundites iga kaadri salvestamise vahel

Open GL

- Clip Alpha (lõika katvus)
 - Lõikab 3D-vaates ära sellest väärtusest väiksema alfaga (läbipaistvus) pikslid.
- Mipmaps (mipmapid)
 - Skaleerib tekstuure 3D-vaate jaoks mipmap-filtri abil. See tõstab pildiesitamise kiirust, kuid nõuab rohkem mälu.
- Anisotropic Filtering (anisotroopne filter)
 - Määrab anisotroopse filtri taseme.
- VBOs
 - Kasutab 3D-vaate joonistamiseks Vertex Bufferi objekte või kui need ei ole toetatud, siis Vertex Array objekte.
- Window Draw Method (akende joonistamise viis)
 - Automatic (automaatne)

Määratakse automaatselt graafikakaardi ja draiveri omaduste põhjal.

Triple Buffer (kolmikpühver)

Kasutab kolmandat ekraanipühvrit, mis vähendab ülejoonistamise vajadust, kuid suurendab mälukasutust.

Overlap (kattuvad)

Joonistab kõik kattuvad alad uuesti. Minimaalne mäluvajadus, kuid rohkem ülejoonistamist.

Overlap Flip (peegelda ülekate)

Joonistab kõik kattuvad alad uuesti. Minimaalne mäluvajadus, kuid rohkem ülejoonistamist (graafikakaartide jaoks, mis toetavad peegeldamist).

Full (täis)

Joonistab iga kord kõik uuesti. Kasuta ainult kontrolliks või siis, kui ükski teine meetod ei tööta.

Text Draw Options (teksti kuvamise sätted)

Luba kasutajaliidese teksti sakisilumine.

Limit Size (piira suurust)

Mälu kokkuhoiuks piirab tekstuuriid varjuse tekstuuride maksimaalset suurust.

Time Out (aegumine)

Aeg sekundites alates viimasest korras, kui OpenGL-i tekstuuri kasutati, enne kui ta vabastatakse. Kui määrad väärtuseks 0, hoitakse tekstuure püsivalt mälus.

Collection Rate (kogumiskiirus)

Aeg sekundites OpenGL-i tekstuuride mäluvabastaja (Garbage Collector) käivitamise vahel.

Sequencer (monteerija)

Prefetch Frames (laadi kaadreid ette)

Kaadrite arv, mis taasesitusel ette renderdatakse.

Memory Cache Limit (mälupehvi piir)

Monteerija mälupehvi maksimaalsuurus (megabaitides).

Solid OpenGL lights (OpenGL-i valgusallikad)

Kui 3D-akna esitusrežiim on Solid View ehk varjutatud, valgustavad stseeni kolm virtuaalset valgusallikat (renderdusel ei ole neid näha). Siin saab muuta nende valgusallikate suunda ja hajus- ning läikevalguse intensiivsust ja värvi.

Miscellaneous (varia)

Color Picker Type (värvivalija tüüp)

Vali, millist tüüpi värvivalijat sa eelistad; seda näidatakse, kui klõpsad LMB  suvalisel värviväljal.

Custom Weight Paint Range (mõjumaalija värvisulatus)

Selle määramiseks, kui palju mingi luu tegelase võret deformeerib, kasutatakse naha kaale (Mesh Skin Weighting). Nende kaalude visualiseerimiseks kasutab Blender värvüleminekut (sinisest roheliseks, siis kollaseks ja siis punaseks). Siin saad sa luua oma värvilahenduse. Selle valiku all olevatel väljadel saad sa lisada, kustutada ja muuta selle värvülemineku värvuspunktide värvi.

Sinu esimene animatsioon kahe pooltunniga. Osa I

Selles peatükis näitame, kuidas animeerida väikest Piparkoogimehikest. Me kirjeldame põhjalikult kõiki samme, kuid eeldame, et oled lugenud [kasutajaliidese](#) peatükki ja saad aru selles raamatus kasutatavatest terminitest ja tähistest.

Selle õppetüki esimese osas valmistame me *staatilise* Piparkoogimehikese. Seejärel, teises osas, paneme ta jalutama.

Note

Palju põhjalikuma sissejuhatuse Blenderisse, rõhuga tegelaskujude animeerimisel, leiad õppetükist

[Blender Summer of Documentation Introduction to Character Animation](#) (See on kirjutatud versioon 2.4 jaoks, kuid ikkagi väga õpetlik).

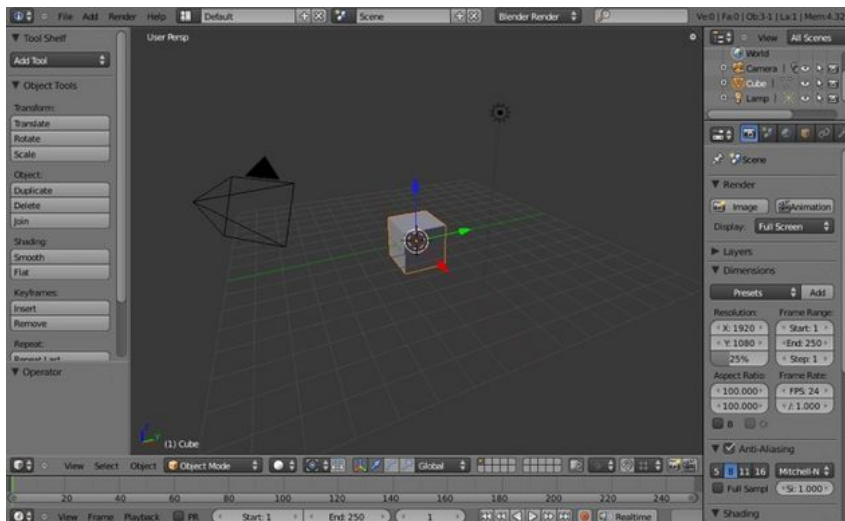


Nagu käesolev õppetükk "Piparkoogimehike Gus", ei nõua ka "BSoD Intro to Character Animation" (BSoD sissejuhatus tegelaste animeerimisse) eelteadmisi Blenderist. See juhatab sind, kuidas luua nullist alustades jalutav ja rääkiv tegelane, ning käsitleb mitmeid võimsaid Blenderi meetodeid, mida siin ei puudutata.

Õppetükki "BSoD Intro to Character Animation" saab ka [PDF-formaadis \(3.75 MB\)](#) alla laadida.

Soojendus

Peale Blenderi käivitamist, peaksid ekraanil nägema vaigestseeni. Keskel olevas 3D-vaates on kaamera, valgusallikas ja kuup. Kuup peaks olema valitud; seda tähistab oranž piirjoon.



Blenderi vaikekuva.

Alustame oma tööala organiseerimisega, paigutades objektid eraldi kihtidele. 3D-stseenid võivad kiiresti muutuda segasteks, kui ekraanil on korraga palju objekte. Kihtide abil saad need objektid, millega sa ei tööta, ära peita ning nad uuesti vajaduse korral nähtavaks teha. Kihid töötavad järgneval viisil:



Kihtide nähtavuse seaded.

Blender annab sulle töö organiseerimiseks kasutada 20 kihti. Seda, millised kihid on hetkel nähtavad, saad näha kahekümnest nupust 3D-akna päisel (nagu pildil *Kihtide nähtavuse seaded*). Sa saad parasjagu nähtavat kihti muuta, klõpsates sellel LMB, ja lülitada erinevate kihtide nähtavust sisse ja välja klõpsates **⇧ Shift LMB**. Nähtavale kihile viitab kihi nähtavuse nuppude sees olev tumedam hall värvus. Viimasest kihist, mis nähtavaks muudetakse, saab aktiivne kiht. Aktiivne kiht on ka see, millele paigutatakse kõik uued loodud objektid.

[Loe kihtide kohta lähemalt »](#)

Teeme kõigepealt ekraani puhtaks, tõstes kaamera ja valgusallika teisele kihile.



Kihile tõstmise dialoog.

Vali kaamera, klõpsates RMB. Siis lisa valikule valgusallikas, klõpsates **⇧ Shift RMB**. Vajuta M ja hiirekursori alla ilmub väike

dialog (nagu pildil *Kihile tõstmise dialog*), mille esimene nupp on sisse lülitatud. See tähendab, et valitud objektid asuvad esimesel kihil. Vali ülemise rea kõige parempoolne nupp. See tõstab kaamera ja valgusallika kihile nr 10.

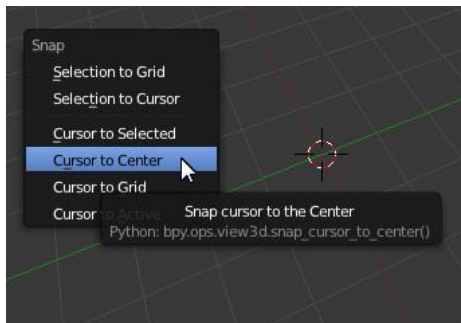
Klaviatuuri kiirvalikud

Blenderis kasutatakse peamiselt suurt hulka kiirvalikuid (ära satu paanikasse - küll sa nendega ära harjud). Enamus neist töötab ainult siis, kui hiirekursor asub vastava paani kohal. Seetõttu ära saa pahaseks, kui kiirvalik M ei tee seda, mida peaks -- liiguta lihtsalt hiir 3D-vaate kohale.

Nüüd veendu, et ainult esimene kiht on nähtav (selle kihi nähtavuse nupu taust on tumedam hall), ja asume modelleerima.

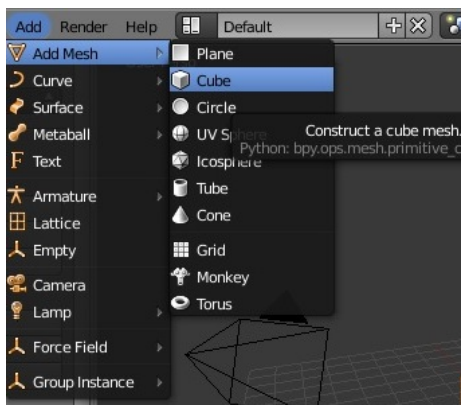
Keha ehitamine

Lülitades kõigepealt sisse numbriluku Num lock, vali eestvaade, vajutades 1 NumPad ja klahviga 5 NumPad paralleelne (*Orthogonal*) projektsioon. 3D-akna ülemine vasaku nurk näitab sulle, kas kasutad paralleelset või perspektiivset projektsiooni.



Naku kursoriga ekraani keskohta

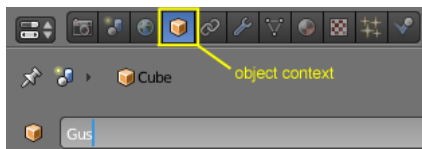
Kui sul ei ole ekraanil kuupi, pead selle lisama. Naku kursoriga ekraani keskohta (0,0,0) menüüga Object » Snap » Cursor to Center või kasutades kiirvalikut ⇧ ShiftS.



Lisa stseenile kuup.

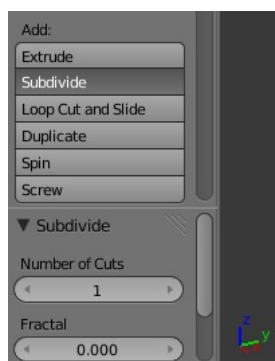
Kuubi lisamiseks kasuta menüüvalikut Add » Add Mesh » Cube (lisa -> võre -> kuup) või kiirvalikut ⇧ ShiftA. Stseeni tekib oranž kuup, mis viitab sellele, et tegemist on aktiivse objektiga. Mine muutmisrežiimi (Edit Mode), vajutades ⇧ Tab.

Muutmisrežiim (Edit Mode) on töörežiim, milles saad võre tippe muuta. Vaikimisi on iga uue objekti kõik tipud valitud (valitud tipud on oranžid, valimata tipud mustad). Objektirežiimis (Object Mode) pole võimalik tippe üksikhaaval valida ega muuta, muuta saab ainult objekti tervikuna. Neid kaht režiimi saab igal ajal nupul ⇧ Tab vajutades vahetada. Hetkel kasutatavat režiimi näidatakse 3D-akna päises.



Gusi ristimine

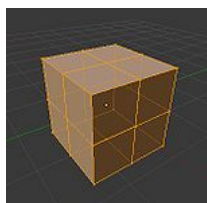
Me anname oma piparkoogimehikesele nimeks "Gus". Selleks tuleb minna objektikonteksti (vaata pilti *Gusi ristimine*), mis asub parempoolses omaduste aknas Properties. Seal on võimalik Gusile esimesel real õige nimi anda.



Osa tööriistariulist

Meie esimene töö on kuubi tippe muutes ehitada Gusi keha. Tööriistad selle jaoks leiab tööriistariulilt (Tool Shelf), mis on 3D-akna osa (ekraani vasakus servas). Kui sa tööriistariulilt (Tool Shelf) ei näe, vajuta lihtsalt kiirklahvi T.

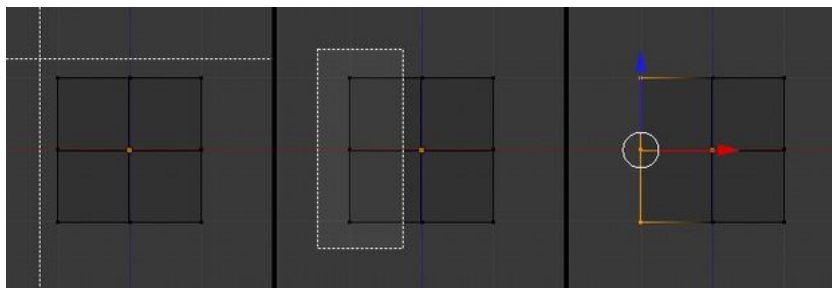
Nüüd otsi tööriistariulilt üles nupp Subdivide (tükelda) (asub nupu Add (lisa) all) ja vajuta üks kord sellele (pilt *Osa tööriistariulist*). See jagab kõik kuubi servad kaheks, luues uued tipud ja küljed. Allpool on näidatud tulemus. Kui tahad samasugust vaadet, siis lülita kiirklahviga Num5 perspektiiv sisse ja pööra vaadet, hoides nuppu MMB ja lohistades hiirt. Ära unusta vaadet tagasi muutmast, vajutades kiirklahve Num1 (eestvaade) ja Num5 (perspektiiv-/paralleelprojektsioon).



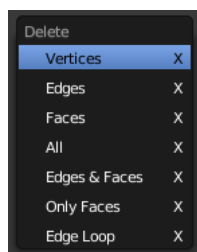
Liiguta hiirekursor 3D-aknasse ja tühistaja valik kiirklahviga A. Tipud muutuvad mustaks.

Õppetükiga jätkamiseks peab sul olema nupp Limit Selection to Visible väljalülitatud.

Nüüd aktiveeri piirdkastiga valik (*Box Select/Border Select*), vajutades B; kursor muutub halliks ristiks. Liiguta kursor ülemise vasaku tipu kohale, hoia LMB all ja lohista hiirega alla ning vasakule nii, et kõik vasaku külje tipud jääksid halli kasti sisse. Nüüd lase LMB lahti (pilt *Piirdkastiga tippude valimine*).



Piirdkastiga tippude valimine.



Kustutamise (Delete) (X) menüü.

Kustuta valitud tipud, vajutades kiirklahvi X ja valides ilmuvast menüüst Vertices (tipud) (pilt *Kustutamise (Delete) (X) menüü*).







Valiku käitumine – Piira valik nähtavatega

Eriti ortograafilises (paralleelprojektsioonis) vaates (mida saab sisse ja välja lülitada kiirklahviga 5 NumPad) võivad osad tipud olla peidetud teiste tippude taha.

Näiteks meie tükeldatud kuubil on 26 tippu, kuid paralleelprojektsioonis eestvaates näed ainult üheksat, sest ülejäänud on varjatud. RMB valib nendest üksteise taga asuvatest tippudest ainult ühe, samas kui piirdkast valib nad kõik. See kehtib siis, kui aktiivsed on kas piirdkasti (Bounding Box) või traatvõre (Wireframe) esitusrežiimid. Kui esitusmeetod on Solid

(vajutatud) või Textured (tekstuuriitud), tuleb nupp Limit Selection to Visible (piira valik nähtavatega) väljalülitada, sest vastasel korral on võimalik valida ainult nähtavaid tippe. (Kui aktiivne on varjatud või tekstuuritud esitusrežiim, leiab nupu Limit

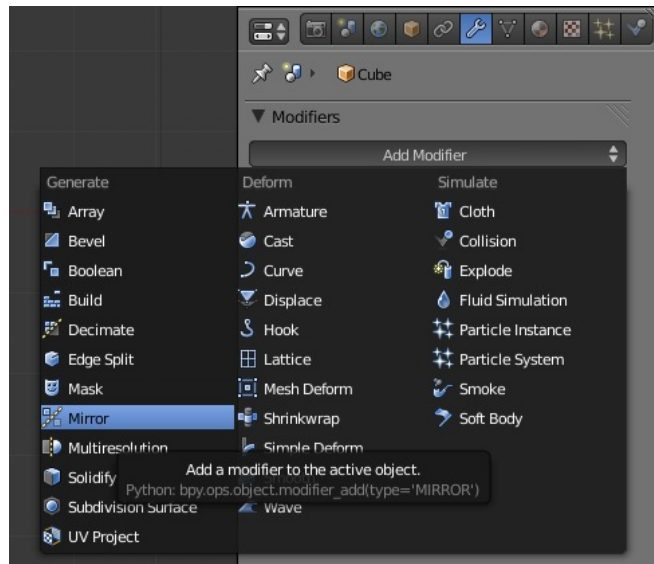
Selection to Visible 3D-akna päisest.)

Teine tööriist tippude valimiseks või valikut eemaldamiseks on ringvalik, mille saab aktiveerida kiirklahviga C. Kui ringvalik on aktiivne, siis nupuga LMB  lohistamine valib tipud. MMB  eemaldab tipud valikust. Hiire ratas muudab valikuringi suurust. Nupp RMB  või kiirklahv  Enter kinnitab valiku ja lahkub ringvaliku režiimist. Kui sa oled tipud kustutanud, nagu eespool näidatud, proovi ka ringvalikut.

Peeglis (*Mirror*) modelleerimine

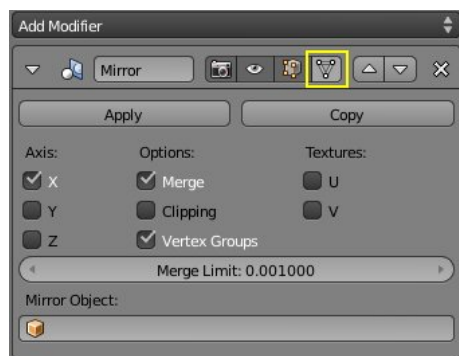
Sümmeetriliste objektide modelleerimiseks saame kasutada peegeldustöötlejat (Mirror modifier). Selle abil võime modelleerida ainult Gusi ühe poole ja Blender loob teise poole ise. Mine omaduste redaktorisse (Properties) ja otsi üles töötlejate (Modifiers) kontekst. (*Töötlejate kontekst*)

[Loe töötlejate kohta lähemalt »](#)



Töötlejate kontekst

Antud hetkel on see päris tühi. Vajutades nuppu Add Modifier (lisa töötleja), ilmub nimekiri, millest vali Mirror (peegeldaja). (*Töötlejate kontekst*)



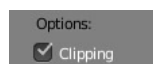
Puurirežiimi (*Cage mode*) nupp.

Lisaks sellele, et töötlejad mõjutavad objekte mitte destruktiivselt, saad mõjutajatega määrata seda, mida nendega töötamise ajal näed. Antud juhul me lülitame sisse puurirežiimi (Cage Mode) nupu, et näha 3D-akna muutmisrežiimis poolläbipaistvaid peegeldatud külgi. (*Puurirežiimi nupp*)

Nüüd valime telje, mida mööda Gusi peegeldatakse, vajutades nuppu X, Y või Z. Peegeldustasapind on selle teljega risti. Meie juhul on selleks X-telg.

Nupp Merge (ühendamine) ühendab kõik peegeldatud tipud, mis asuvad samal kohal või millede omavaheline asukoht on lähem kui ühendamise piiri sätte Merge Limit väärtus. Kõik tipud, mis on peegelduspinnale lähemal kui määratud väärtus, liigutatakse täpselt peegelduspinnale ja ühendatakse vastava samas kohas oleva tipuga. Piirväärtuseks saab määrata **0.000** kuni **1.000** Blenderi ühikut – selle suurus sõltub töö iseloomust ja objektide üldsuurusest.

Gusi modelleerimisel on märgata, kui tipud asetsevad peegelduspinnast rohkem kui **0.1** ühiku kaugusel, väiksema vahemaa puhul aga mitte. Et vältida meie võre keskele suure rebendi tekkimist või seda, et mõni veidi nihkes tipp kogemata kahe silma vahele ei jääks, määrame seade Merge Limits (ühendamise piir) väärtuseks **0.1**.




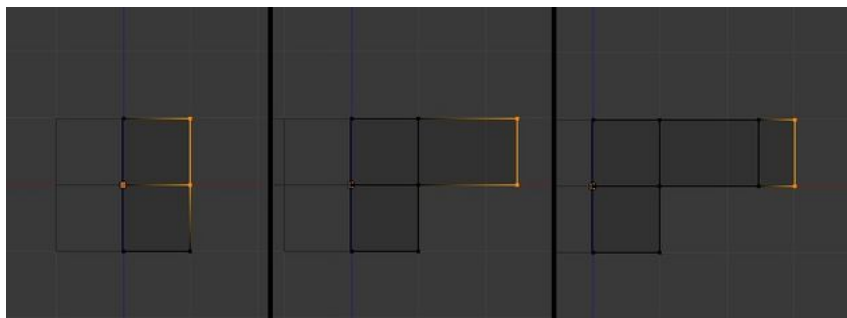
Kärpimisnupp (*Clipping*).

Lõpuks, lülitades sisse nupu Clipping (kärpimine) (pilt *Kärpimisnupp*), muutub peegelduspind piiriks, üle mille tippu nihutada ei saa. Lisaks, kui Clipping (kärpimine) on aktiivne, kleepub iga peegelduspinnal **asuv** tipp selle külge.

Nagu näed, on peegeldajal hulk seadeid, mis muudavad meie elu lihtsamaks.

Käed ja jalad

Teeme nüüd Gusile käed ja jalad. Kasutades just õpitud meetodit, vali piirkastiga (Box Select) kaks kõige ülemist tippu (pilt *Käte eendamine kahe sammuga*). Reaalselt valib see ka ülejäänud neli nende taga, seega kokku kuus tippu. Vajuta kiirklahvi E, et need eendada (või kasuta nuppu Extrude Region (eenda piirkond) tööriistariiulis). Sellega tekivad uued tipud ja küljed, mida sa saad hiirega liigutada. Nihuta need poolteist ruutu paremale ja kinnita siis nende asend, klõpsates LMB . Korda eendamist, vajutades uuesti E, ja nihuta uued tipud veel poole ruudu võrra paremale. Pildil (*Käte eendamine kahe sammuga*) on näha tegevuste järjekord.



Käte eendamine kahe sammuga.

Tühistamine ja uuesti tegemine (Undo/Redo)

Blenderil on kaks tegevuste tühistamise puhvrit: üks Muutmisrežiimis (Edit Mode) ja teine Objektirežiimis (Object Mode).


Tühistamiseks vajuta muutmisrežiimis CtrlZ. Korda CtrlZ vajutamist, et minna tegevuste ajaloos nii kaugele, kui tühistamispuhver lubab. Kiirklahv ⇧ ShiftCtrlZ teeb tühistatud toimuingu uuesti.

(Macil kasuta Ctrl asemel klahvi ⌘ Cmd.)


Meeles tuleb pidada kahte asja:

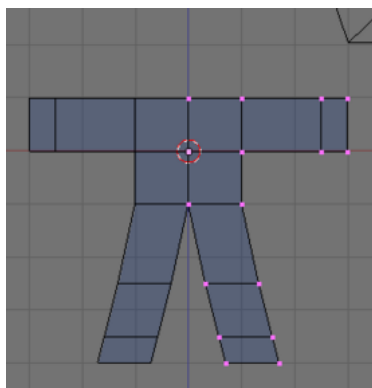
- Muutmisrežiimis tühistamine töötab ainult sellel objektil, mis on hetkel muutmisrežiimis.
- Tühistamispuhver ei lähe kaduma kui sa muutmisrežiimist lahkud, kuid nullitakse niipea, kui sa hakkad *teist* objekti muutmisrežiimis redigeerima.

Objektirežiimis töötavad samad kiirklahvid. CtrlZ tühistab, ⇧ ShiftCtrlZ teeb uuesti. Kui oled redigeerinud objekti muutmisrežiimis ja selle objekti tühistamispuhver ei ole nullitud, tühistab CtrlZ vajutamine Objektirežiimis (kui tühistamiste järjekord on jõudnud sinnani) kõik need muudatused korraga.

Kui sa poole tegevuse pealt ümber mõtled, saad käimasoleva tegevuse paljudel juhtudel tühistada, vajutades Esc või RMB .


Kokkulangevad tipud

Eendamisel luuakse kõigepealt uued tipud ja seejärel nihutatakse neid. Kui mõtled nihutamise ajal ümber ja vajutad tühistamiseks Esc või RMB , jäävad uued tipud vanade kohale alles! Lihtsaim viis minna tagasi olukorrani enne eendamise algust on tühistada eendamine, vajutades (CtrlZ). Mõnikord on kasulik luua uued tipud just sellel viisil ning seejärel nihutada, skaleerida või pöörata neid klahvidega G, S või R.



Body.

Gusil peaks nüüd olema sinu modelleeritud vasak käsi (ta on näoga meie poole) ja Blenderi poolt lisatud parem käsi. Me loome nüüd tippe samal viisil kolm korda eendades vasaku jala. Ürita teha midagi sarnast nagu paremal oleval pildil (*Keha*). Kui kasutad nuppu *Extrude - Region* (eenda piirkond), pead ajutiselt Mirror (peegeldaja) töötleja välja lülitama, eemaldades linnukese jaotuse Axis (telg) valikut x ja pärast väljasurumise lõppu jälle sisse lülitama (vastasel korral tekib Gusile pükste asemel seelik).

Sa saad eendatavate tippude suuna vabastada, klõpsates MMB  (pärast E vajutamist, kuid enne, kui sa klõpsad LMB ). Kui sa seda ei tee, suunduvad mehikese jalad otse alla, mitte alla ja kõrvale nagu pildil (*Keha*).

Vihje: kui tahad asju täpselt paigutada, hoia nihutamise ajal all klahvi Ctrl.


Sellega on peegeldusega modelleerimine lõppenud. Järgmistes sammudes eksperimenteerime teiste tehnikatega. Peame muutma mudeli parema poole *tegelikuks*, sest miski, mida me töötlejatega teeme, ei ole jääv, enne kui me pole me töötlejat *rakendanud* (*apply*). Mine Gusiga objektirežiimi (Object Mode) (kui ta on ikka veel muutmisrežiimis, vajuta ⇧ Tab) ja klõpsa töötlejal Mirror

Modifier (peegeldaja) nuppu Apply.

Pea

Gusile on vaja ka pead.



Mine uuesti muutmisrežiimi (Edit Mode); vajuta ⇐ Tab.

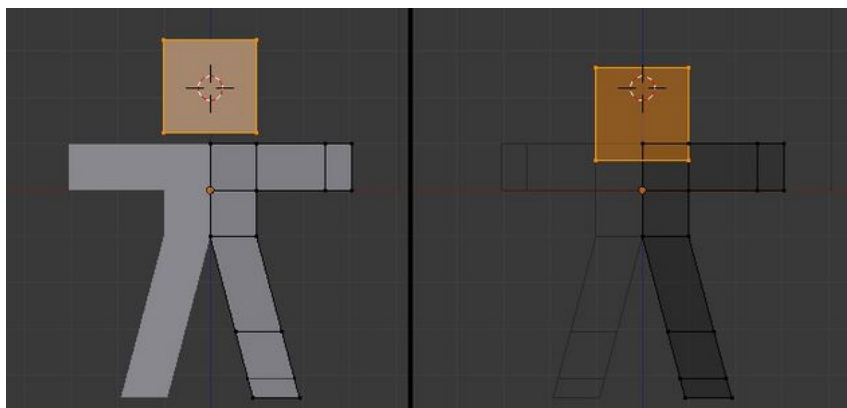
Liiguta hiirekursor Gusi kehast täpselt ühe ruudu võrra ülespoole. Selleks, et paigutada kursor täpselt ruudustiku punktile, paiguta ta alguses hiirenupuga LMB  võimalikult õige koha lähedale ja ava siis kiirklahvi ⇧ ShiftS vajutades Snap Menu (nakkumismenüü). Valik Cursor to Grid (kursor ruudustikule) paigutab kursori täpselt ruudustiku lähimale punktile. Seda tahamegi praegu teha. Valik Cursor to Selection (kursor valikule) paigutab kursori valitud objekti keskmesse, mis võib mõnikord kasulikukuks osutuda.

Lisa Gusi pea jaoks uus kuup (vajuta ⇧ ShiftA >> Add >> Cube) (lisa -> kuup) (joonise *Pea lisamine* vasakpoolne pilt).

Objektide loomine

Kui sa lisad objekti muutmisrežiimis teisele objektile, muutub lisatud uus objekt olemasoleva osaks. Seega, lisades selle kuubi ajal, kui Gusi keha on muutmisrežiimis, saab sellest automaatselt tema keha osa.

Nüüd vajuta G, et minna nihutamisrežiimi (Grab Mode), ja liiguta uut loodud kuupi allapoole. Sa saad piirata liikumise üles-alla vertikaaljoonega, nihutades pead veidi allapoole ja vajutades siis MMB . Nihuta Gusi pead umbes kolmandiku võrgusammu võrra allapoole ja fikseeri siis tema asend, klõpsates LMB  (joonise *Pea lisamine* parempoolne pilt).



Pea lisamine.

Pinnatükeldaja (*Subdivision Surface / Subsurf*)



Pinnatükeldaja

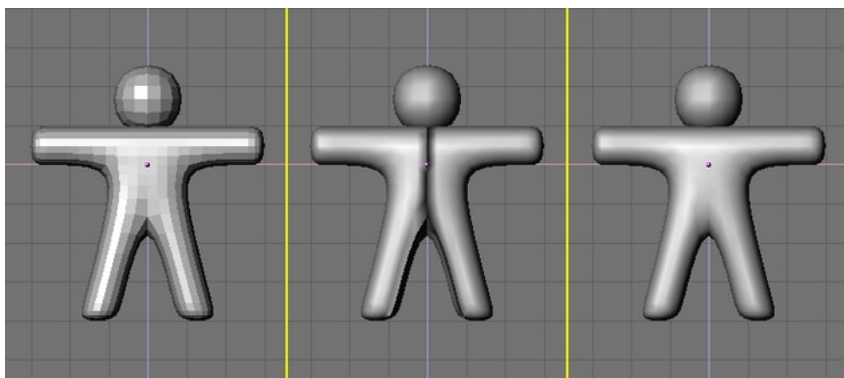
Järgmiseks sammuks pead valima terve Gusi, mitte ainult tema pea (vajuta A – võimalik, et seda tuleb teha kaks korda).

Meie senise töö tulemus on parimal juhul nurgeline figuur. Et seda siluda, otsi üles Töötlejate Modifiers kontekst ja lisa sinna Subdivision Surface Modifier (pinnatükeldaja), (pilt *Pinnatükeldaja*). Veendu, et numbriväljade View (vaade) ja Render (renderdus) (mõlemad on alljaotuses Subdivisions (tükeldusastmed)) väärtus oleks 2 või vähem. View (vaade) määrab tükeldustaseme, mida sa näed 3D-vaates; Render (renderdus) määrab tükeldusastme renderdamisel.

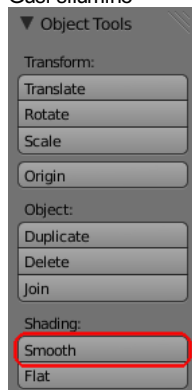
Pinnatükeldus (Subdivision Surface)

Pinnatükeldus on innovaatorlik modellerimisvahend, mis silub dünaamiliselt konarlikku võret. Tema tööpõhimõte on teha esialgselt võrest palju tihedam versioon ning nihutada selle tippe nii, et nad järgivad sujuvalt esialgse rohmaka võre kuju. Objekti üldine kuju on ikka määratud esialgse kandilise võre tippude poolt, kuid renderdatud pilt on on sile ja ühtlane.

Et Gusile pilku heita, mine muutmisrežiimist välja (⇐ Tab) ja vali Solid (varjutatud) esitusviisi, vajutades Z (kui see pole juba aktiivne). Ta peaks nägema välja nagu pildil (*Gusi silumine*).




Gusi silumine

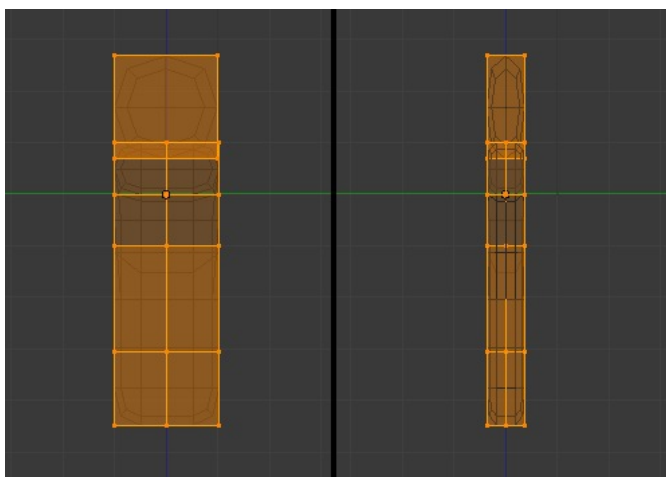


Objektitööriistad.

Et teha Gus veelgi siledamaks, vajuta nuppu Smooth (silendamine) 3D-akna paneelil Tool Shelf (kiirklahv T alajaotuses Shading (varjutamine)). Gus muutub nüüd siledaks, ehkki tema keskkohas võib olla näha veidraid musti jooni. Töötleja Mirror Modifier (Peegeldaja) kasutamine aitab seda enamasti vältida, kuid nii võib juhtuda, kui kasutada eendamist ning peegeldamist, nagu tehti enne peegeldaja lisamist Blenderisse (pilt *Gusi silumine*, keskel). Need jooned tekivad sellest, et pinnatükeldaja tihe võre arvutatakse kandilise võre normaali suundi arvestades (pinnanormaal on küljega risti ja määrab, kummas suunas külg "vaatab"), mis ei pruugi kõik olla õigepidi (mõned küljenormaalid võivad olla suunatud väljapoole, mõned sissepoole). Et normaale algväärtustada, mine tagasi muutmisrežiimi (Edit Mode), vajuta \leftrightarrow Tab, vali kõik tipud (A) ja vajuta kiirklahvi CtrlN. Nüüd peaks Gus olema ilus ja sile (pilt *Gusi silumine*, paremal).

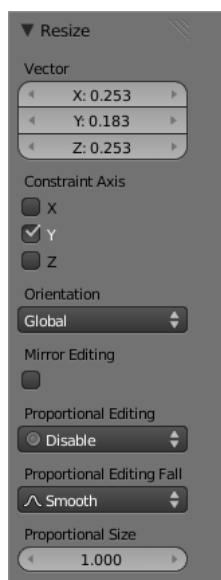
Hoia MMB  ja lohista hiirega, et Gusi kõigist suundadest vaadelda. Appi, ta on liiga paks!

Piiratud skaleerimine






Gusi õgvendamine piiratud skaleerimisega

Teeme Gusi kõhnemaks:



Viimase tegevuse seaded paneelil Tool Shelf (tööriistariil)

- Kui sa seal veel muutmisrežiimis ei ole, siis mine sellesse (\leftrightarrow Tab), seejärel tagasi traatvõre esitusrežiimi (Z). Lülita sisse külgvaade Num3 ja vali kõik tipud kiirklahviga A. Kui sulle meeldib, võid järgnevaid samme teha ka objektirežiimis (Object Mode).
- Vajuta S ja liiguta hiirt horisontaalsuunas. (Klõpsa MMB , et piirata skaleerimine ainult ühe teljega, või vajuta sama tulemuse saamiseks klahvi Y). Kui sa nüüd liigutad hiirt Gusi poole, peaks ta muutuma kõhnemaks, kuid jääma sama pikaks.
- 3D-akna päisel on näha skaleerimisfaktor. Vajuta ja hoia all klahvi Ctrl: skaala muutub nüüd astmeliselt: 0,1 kaupa. Skaleeri Gus kõhnemaks, kuni skaleerimisfaktor on 0.2 ja kinnita siis suurus, klõpsates LMB . Kui see viimane teisendus läks valesti, saad sa ikka veel tema parameetreid muuta. Need on nüüd näha paneeli Tool Shelf (tööriistariil) allosas (vaata pilti *Viimase tegevuse seaded tööriistariilil*).
- Vali uuesti eestvaade (Num1) ja varjutatud esitusrežiim (Z) ning pööra siis oma vaadet MMB  abil. Gus näeb nüüd palju parem välja!


Vaatame, milline Gus välja näeb

Me hakkame kohe tegema oma esimest renderdust, kuid enne on meil veel veidi tööd teha.

- Kui sa veel ei ole, siis mine objektirežiimi (Object Mode, \leftrightarrow Tab).

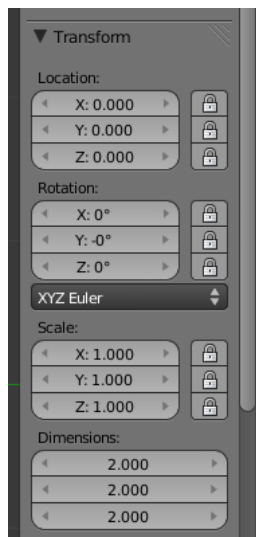


Kihtide 1 ja 10 korraga nähtavaks tegemine.

- Klõpsa \div Shift LMB  kihtide nähtavuse nuppude (asuvad 3D-akna päisel) ülemisel paremal nupukesel (pilt *Kihtide 1 ja 10 korraga nähtavaks tegemine*), mis teeb nähtavaks nii kihi 1 (Gusi kiht) kui ka kihi 10 (kiht, millel asuvad kaamera ja valgusallikas).



Vihje

Meenutame, et viimasena valitud kiht on aktiivne ja seega kõik järgnevad tegevused toimuvad kihil 10.





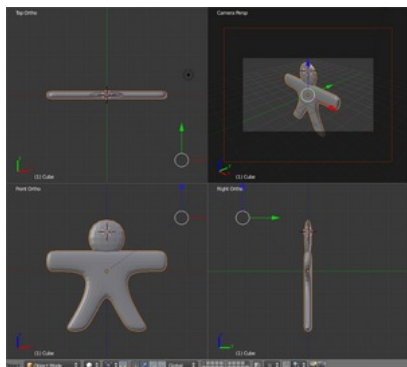
Teisenduste paneel

- Vajuta N, mis näitab paneeli Properties (omadused), ja otsi sealt paneel Transform (teisendus) (pilt *Teisenduste paneel*). Objekti asukoht on määratud X, Y ja Z väärtustega.

**Vali kaamera (RMB ) ja nihuta ta asukoha (x=7, y=-10, z=7) kanti. Tee seda, vajutades G ning kaamerat lohistades. Sul võib olla kõigi kolme koordinaadi muutmiseks vaja vahepeal vaatesuunda muuta ning kaamerat teist korda nihutada. Kui eelistad objekti asukohta määrata tema numbriliste koordinaatide sisestamisega, saad seda teha, klõpsates vastaval väljal LMB ) ja sisestades siis vajaliku väärtuse.

Kaamera seadistamine

Selleks, et kaamera Gusile suunata, jäta kaamera valituks ja lisa valikule Gus, klõpsates  Shift RMB . Kaamera peaks muutuma tumeoranžiks (valitud) ning Gus heleoranžiks (valitud ja aktiivne). Nüüd vajuta CtrlT ja vali menüüst TrackTo Constraint (jälgimispiiraja). See sunnib kaamerat Gus jälgima, pöörates end alati tema suunas. See tähendab, et võid liigutada kaamera kuhu iganes tahad, olles kindel, et Gus püsib alati vaatevälja keskel.




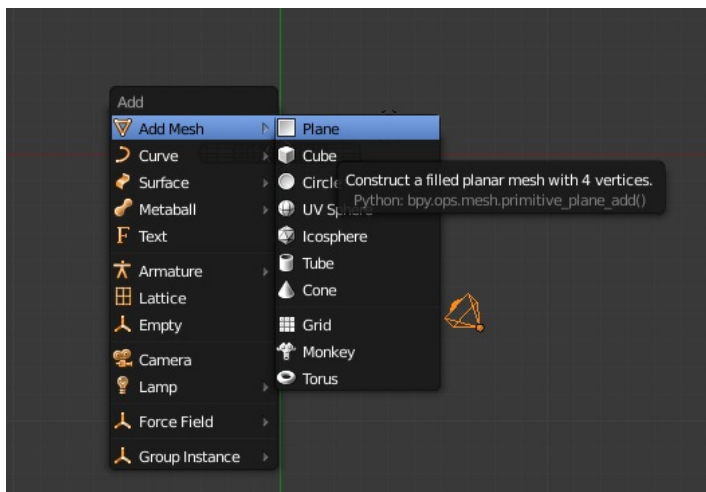
Kaamera asukoht Gusi suhtes.

(pildil *Kaamera asukoht Gusi suhtes*) on näha Gusi pealt-, eest-, küljelt- ja kaameravaade. Selleks, et lülitada sisse kaameravaade, vajuta 0 NumPad või vali menüüst View>>Camera (Vaade >> Kaamera). Selleks, et saada ekraanile nelikvaadet (Quad View) nagu pildil, vajuta CtrlAltQ või vali menüüst View>>Quad View (Vaade >> Nelikvaade).

Maapind

Nüüd me peame tegema maapinna, millel Gus seisab.

- Pealtvaates (7 NumPad või menüüst View>>Top (Vaade >> Pealt)) ja objektirežiimis
- Lisa stseenile tasand ( ShiftA või menüüst >>Add>>Mesh>>Plane (Lisa >> Võre >> Tasand)).



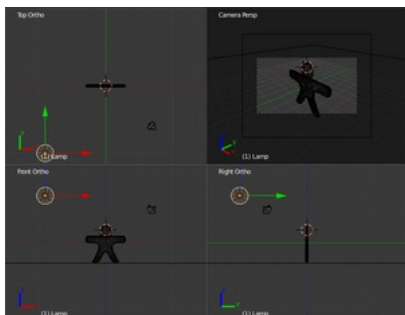
Märkus

On oluline, et sa ei oleks muutmisrežiimis, vastasel juhul saab uuest lisatud objektist olemasoleva aktiivse objekti osa, nii nagu me lisasime Gusi pea.

- Lülita sisse eestvaade (1 NumPad või View>>Front (Vaade >> Eest)) ja nihuta (G) tasand Gusi jalgade alla, hoides samal ajal asendi säilitamiseks Gusi suhtes all klahvi Ctrl .
- Mine kaameravaatesse (0 NumPad või View>>Camera (Vaade >> kaamera)) ja samas kui tasand on ikka veel valitud, vajuta S, et seda skaleerida.
- Suurenda tasandit nii palju, et tema ääred ulatuksid kaamera nägemisväljast (mida märgib heledam piirkond kaameravaate keskel) üle.

Valgusallikad

Nüüd lisame stseenile veidi valgust!



Valgusallika lisamine

- Liiguta pealtvaates (7 NumPad) olemasolev lamp paremale (kui su stseenis ei ole valgusallikat, saad selle lisada, vajutades \star ShiftA >>Add>>Lamp>>Lamp (lisa -> lamp -> lamp)) Gusi ette, kuid kaamerast teisele poole, näiteks koordinaatidele (x=-9, y=-10, z=7) (*Valgusallika lisamine*).



Valgusallika objektandmete nupp

- Kui valgusallikas on objektirežiimis valitud, klõpsa omaduste aknas objektandmete (Object Data) ikooni (meenutab väikese päikese pilti). Sa näed valgusallika alammenüüd valikutega Point (punkt), Sun (päike), Spot (kohtvalgus), Hemi (poolkera) ja Area (ala). (Pilt *Valgusallika objektandmete nupp*).



Kohtvalguse seadistused

- Vajuta omaduste akna paneelil Lamp valikunuppu Spot (kohtvalgus), et muuta lamp kohtvalguse allikaks (pilt *Kohtvalguse seadistused*), ja anna sellele kahvatukollane värv (R=1, G=1, B=0.9), vajutades valgele nupule (mis on tegelikult värvimääraja, mis avab värvivaliku akna). Määra paneelil Spot Shape (valgusvihi kuju) välja Size (suurus) väärtuseks umbes 40 ja Blend (sulandamine) väärtuseks 1.0.
- Pane see kohtvalguse allikas Gusi jälgima samamoodi, nagu tegid kaameraga: vali valgusallikas, vajuta alla ⇧ Shift, vali Gus ja seejärel CtrlT>>TrackTo Constraint (jälgimispiiraja).
- Lisa teine lamp, mis annab ühtlasemat taustavalgust (⇧ ShiftA >>Add>>Lamp>>Hemi (Lisa >> Lamp >> Poolkera)). Määra välja Energy (energia) väärtuseks 0.2 (pilt *poolkeravalguse seaded*). Nihuta ta veidi kaamerast kõrgemale (x= 7, y= -10, z=9) ja pane ka tema Gusi jälgima.

Kaks valgusallikat?

Kasuta pehme ja realistliku valguse saamiseks kahte või enam valgusallikat, sest tegelikkuses ei lähtu valgus kunagi ainult ühest punktist.

Renderdamine



Renderduskonteksti nupp

Me oleme peaaegu valmis renderdamiseks. Esimese sammuna vajuta omaduste (Properties) akna päisel renderduskonteksti nuppu (pilt *Renderduskonteksti nupp*).



Renderduskontekst

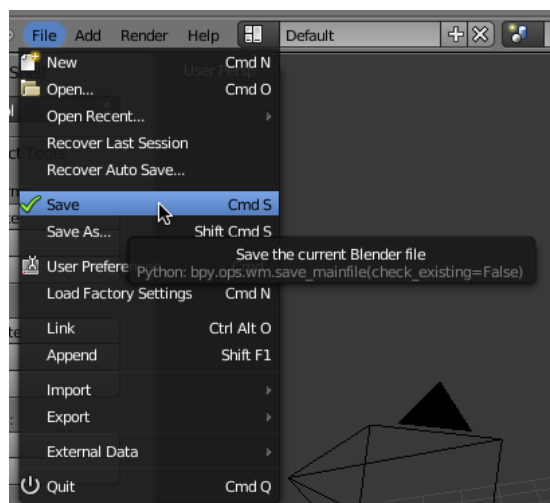
Kasutame vaikimisi renderduse seadeid, nagu näha pildil (*Renderduskontekst*).

Nüüd vajuta nuppu Image (pilt) või kiirklahvi F12. Tulemus, mis on näha pildil (*Palju õnne! Sinu esimene renderdus*) ei ole eriti ilus. Meil on vaja materjale ja palju detaile, nagu silmad ja nii edasi.



Palju õnne! Sinu esimene renderdus.

Töö salvestamine



Salvestusmenüü

Kui sa seda juba teinud ei ole, on nüüd sobiv aeg oma töö salvestada, valides menüüst File>>Save (Fail >> Salvesta), nagu näha pildil *Salvestusmenüü*, või vajutades kiirklahvi Ctrl+S. Blender hoiatab sind, kui püüad olemasolevat faili üle kirjutada.

Blender teeb ka automaatseid varukoopiaid sinu süsteemi ajutiste failide kataloogi. Vaikimisi juhtub see iga viie minuti tagant ja faili nimeks on mingi number. Nende varukoopiate laadimine on teine meetod ebavajalike muutuste tühistamiseks.

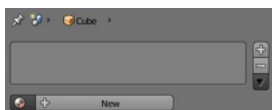
Materjalid ja tekstuurid

On aeg anda Gusile kena küpsisesamane materjal.



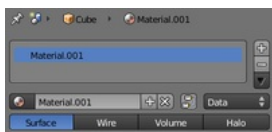
Materjalikonteksti nupp.

- Vali Gus. Siis vali materjalipaneeli avamiseks omaduste akna päisel materjalikonteksti nupp (pilt *Materjalikonteksti nupp*).



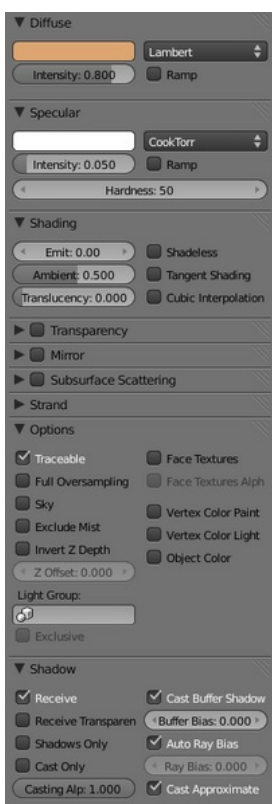
(Peaaegu) tühi
materjalikontekst

- Omaduste aken on peaaegu tühi, sest Gusil ei ole veel materjali. Materjali lisamiseks vajuta materjalipaneelil (Material) nupule + New(uus) (pilt *(Peaaegu) tühi materjalikontekst*).



Materjalikonteksti ülemine
osa

- Omaduste aken täitub nüüd paneelide ja nuppudega. Materjalide nimekirja ja unikaalse andmebloki koodi (*Unique Datablock ID*) väljale ilmuvad tekstid materjali nimega (eeldatavasti on see "Material.001"). Klõpsa viimatimainitud väljal (näha pildi Materjalikonteksti ülemine osa alumises osas) ja kirjuta sinna mingi tähenduslik nimi, näiteks "GingerBread" (piparkook) (ära jutumärke kirjuta).



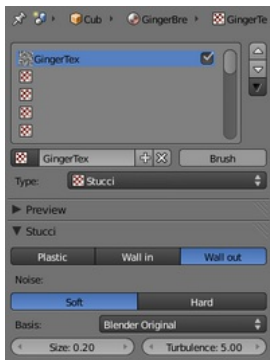
Esimene piparkoogimaterjal.

- Muuda vaikeseadeid, nagu näidatud pildil (*Esimene piparkoogimaterjal*); see annab esialgse materjali. Pane tähele, et pead mõned paneelid avama, klõpsates LMB väikestel kolmnurkadel nende päisel.



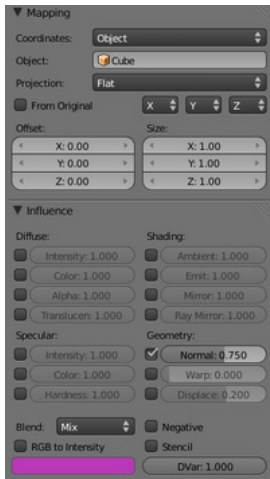
Tekstuurikonteksti nupp

- Vajuta akna Properties päisel tekstuurikonteksti nuppu (pilt *Tekstuurikonteksti nupp*) ja seejärel vali Add new (lisa uus). Lisame esimesse kanalisse tekstuuri. Anna sellele nimeks "GingerTex."



Tekstuurikontekst

- Vali menüünupust Type (tüüp) esialgse väärtuse Clouds (pilved) asemel Stucci (krohv) ja täida väljad nii, nagu pildil (*Tekstuurikontekst*) näidatud.



Krohvitekstuuri seaded

- Täida tekstuurikonteksti paneelidel Mapping (laotamine) ja Influence (mõju) väljad nii, nagu pildil (*Krohvitekstuuri seaded*): eemalda linnuke kastist Color (värv) ja lisa kasti Normal (normaal). Seejärel sea numbrivälja Normal (normaal) väärtuseks **0.75**. Need seaded teevad meie krohvitekstuuri "faktuuriakaardi" ja muudavad Gusi küpsisele samasemaks.



Täiendava müratekstuuri seaded

- Nüüd vali tekstuurikonteksti tekstuuride nimekirja teine rida ja lisa järgmine tekstuur. Pane sellele nimeks "Grain" (teralisus) ja seadista vastavalt pildile (*Täiendava müratekstuuri seaded*). Tekstuur ise on lihtne protseduuriline tekstuur Noise.



Väga lihtne maapinna materjal

- Lisa maapinnale sobiv materjal, näiteks tumesinine, mis on näidatud pildil (*Väga lihtne maapinna materjal*). Võid vabalt valida sulle meeldiva sinise varjundi.

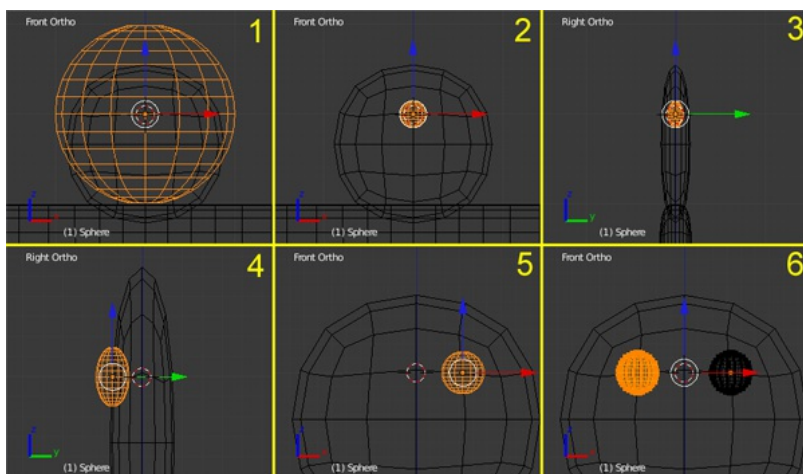
Silmad ja detailid

Viimistluseks lisame oma mehikesele silmad ja mõned detailid.



Kihtide nähtavus

- Kõigepealt tee esimene kiht ainsaks nähtavaks kihiks, klõpsates LMB tema nupul (pilt *Kihtide nähtavus*). See peidab valgusallikad, kaamera ja maapinna.
- Paiguta kursor Gusi pea keskele. (Pea meeles, et töötad 3D-keskkonnas, seega kontrolli paigutust vähemalt kahes vaates!)
- Lisa objektirežiimis stseenile kera (ShiftA >>ADD>>Mesh>>UVsphere (Lisa >> Võre >> UV kera)). Vajuta klahvi F6 ja määra segmentide (meridiaanide) arvaks 16. Tulemus on näidatud pildi *Silmade tegemine* osal 1.
- Skaleeri kera väiksemaks (S), kuni skaalafaktori väärtuseni umbes **0.15** kõigis mõõtmetes. Seejärel lülita sisse külgsuuna (3 NumPad) ja skaleeri objekti ainult horisontaalsuunas (Y) väärtuseni **0.5** (vaata pildi *Silmade tegemine* osasid 2 ja 3).

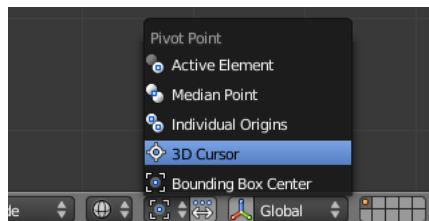


- Kui vaja, suurenda veidi vaadet + NumPad, Wheel või Ctrl MMB ja lohista siis kera (G) vasakule, kuni ta on poolenisti pea sees (pildi *Silmade tegemine* osa 4).
- Pöördu tagasi eestvaatesse (1 NumPad) ja nihuta kera külgsuunas paremale. Paiguta ta kohta, kus peaks olema Gusi silm (pildi *Silmade tegemine* osa 5).

Koopia peegeldamine üle kursori

- Mine muutmisrežiimi (Tab). Vali 3D-akna päiselt sihikuga teisenduskeskme nupp (*Pivot point: 3D Cursor* - teisenduskeske: 3D kursor) (3D Transforms (3D-teisendaja) hüpab selle tulemusena keralt kursori asukohta). Kõik silma tipud peaksid olema valitud (kui mitte, siis vajuta A, et nad kõik valida). Nüüd tee valitud tippudest duplikaadid, vajutades kiirklahvi ShiftD või nuppu

Duplicate (duplikaat) tööriistariilil (Tool Shelf) ja seejärel Esc, et katkestada nende paigutamine hiirt liigutades.



Teisenduskeske: 3D kursor

- Vajuta CtrlM, et peegeldada valitud tipud, seejärel X, et peegeldada piki X-telge, ja siis LMB või ↵ Enter, et peegeldus kinnitada. Sea teisenduskeske tagasi esialgsesse asendisse (Median Point (keskpunkt)). Tulemus on näha pildi *Silmade tegemine* osal 6.

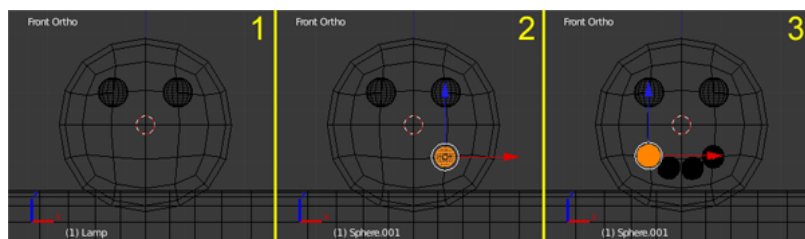
Peegeldamine

Kiirklahviga CtrlM saab peegeldada ka objektirežiimis.

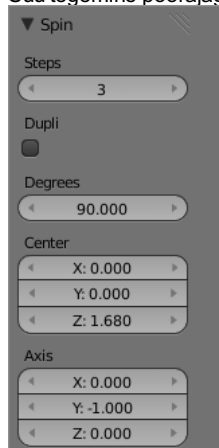
Nüüd on Gusil kaks silma.

Suu

- Lahku muutmisrežiimist (⇌ Tab) ja paiguta kursor (tuleta meelde klahvi ⇧ ShiftS) Gusi näo keskpunktile nii lähedale kui suudad. Lisa uus kera, vähenda tema suurust ja nihuta teda – täpselt nii, nagu tegid silma puhul, selle vahega, et tee ta veidi väiksemaks (suuruseks 0.1 mitte 0.15). Paiguta ta kursorist alla paremale, tükeldatud pinna ühe tipu kohale, nagu näidatud pildi *Suu tegemine pööräjaga* osas 2.



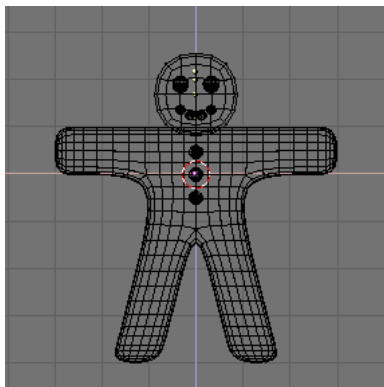
Suu tegemine pööräjaga



Pööraja valikud
tööriistariilil

- Mine muutmisrežiimi (⇌ Tab). Nüüd vajuta kiirklahvi AltR või nuppu Spin (pööra) tööriistariilil. Tekib mitu koopiat kerast.
- Tööriistariili allosas (vajuta F6, kui see ei ole näha) määra pöörde detailid: välja Degr (kraade) väärtuseks **90** ja välja Steps (samme) väärtuseks **3**. Tulemuseks peaks olema Gusi suu, nagu näidatud pildi *Suu tegemine pööräjaga* osas 3.

(TOIMETAJA MÄRKUS: Viimase käsu omadused (üaltpoolt näites *Spin*) ei ole alati 3D-aknas näha (vähemalt mitte toimetaja Blenderis (V2.5 beta, Mac)), nii et ära satu paanikasse – kusagil seal nad ikkagi on (vähemalt vaikimisi ekraanipaigutusel))



Valmis Gus!

- Nüüd mine tagasi objektirežiimi ning lisa Gusi nööpideks veel kolm kera (peast allapoole piki Z-telge). Kui oled ühe nööbi valmis teinud, võid lihtsalt muutmisrežiimist lahkuda, luua klahviga ⇧ ShiftD sellest koopia ja nihutada selle õigesse kohta, nagu pildil *Valmis Gus!*.

Kerade kinnitamine

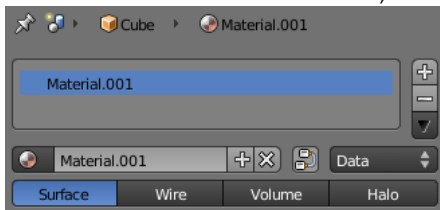
Kui me tahame, et me saaks haarata Gusist kinni ja liigutada teda ühe tervikuna (see puudutab ka õppetüki teises osas käsitletavat animeerimist), peame me liitma need väikesed kerad, mis moodustavad silmad, suu ja nööbid, Gusi kehaga. Mine objektirežiimi ja vajuta A, kuni valik on tühi. Nüüd klõpsa parema hiirenupuga ühele kerale (sobib ka, kui mitu tükki on grupina valitud). Hoides all klahvi ⇧ Shift, vali keha. Vajuta kiirklahvi CtrlP ja vali seejärel menüüst Object. Tühista valik ning liida seda tegevust korrates ka kõik teised elemendid kehaga.

Silmade materjal

Anname silmadele šokolaaditaolise materjali. Suu materjaliks paneme valge suhkrutaolise materjali ja anname nööpidele värvidega pisut särtsu. Tuleta meelde, mida sa tegid Gusi endaga ja ole loominguline – ning varsti on Gus super-piparkoogimoodne.

Materjali jagamine objektide vahel

Selleks, et anda ühele objektile sama materjal mis teisele (näiteks selleks, et kasutada suu valget materjali ühel nööbil), vali see materjal **materjalimenüü** nimekirjast, mis ilmub, kui sa vajutad nuppu Browse ID Data (sirvi ID andmeid), mis asub omaduste akna materjalikontekstis, välja Data Block ID Name (andmebloki ID nimi) kõrval (pildil *materjalikontekst* on see, mida sa vajad väike ruuduline kera teksti *Material.001* kõrval).



Renderdamine

Kui oled materjalidega lõpetanud, tee kiht **10** jälle nähtavaks (mäletad, kuidas see käis? Vihje - vaata 3D-akna päisele), nii et välja ilmuvad valgusallikad ja kaamera, ning siis renderda uuesti (F12).

Tulemus peaks rohkemal või vähemal määral nägema välja nagu pildil (*Liikumatu renderdus valmis Gusist!*).



Liikumatu renderdus valmis Gusist.

Salvestamine

Salvesta oma pilt kiirklahviga F3 aknas UV/Image Editor (UV/Pildiredaktor), kus näidatakse valmis renderdatud tulemust. Anna oma pildile failisirvijas nimi, vali salvestuskataloog ning salvesta.

Saad valida pildiformaadi (JPEG, PNG ja nii edasi) tööriistariulilt failisirvija parempoolses servas.

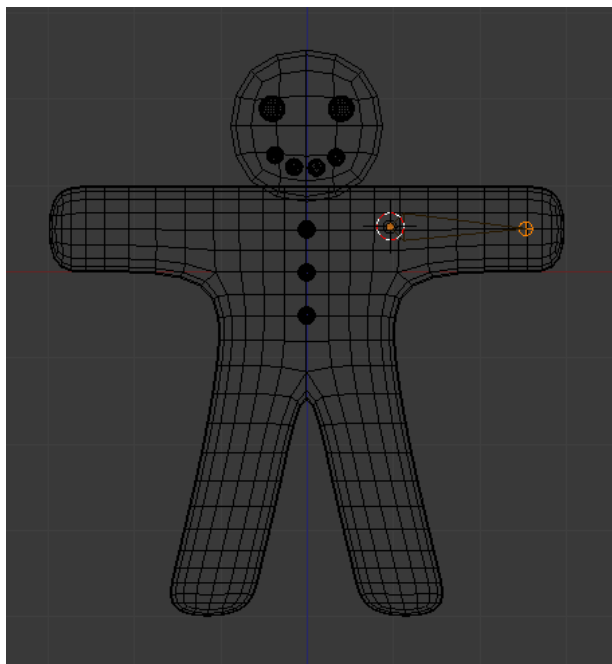
Blender lisab failinimele laiendi, nagu oled ilmselt harjunud (see on versioonis 2.5 uus – versioonis V2.4x pidid sa seda käsitsi tegema).

Sinu esimene animatsioon kahe pooltunniga. Osa II

Kui me sooviksime tulemuseks seisvat pilti, oleks meie senine töö piisav. Kuid me soovime, et Gus liiguks! Järgmise sammuna anname talle luustiku ehk skeleti, mis paneb ta liikuma. Seda kutsutakse taageldamise peeneks kunstiks. Gusil saab olema väga lihtne taageldus: neli jäset (kaks kätt ja kaks jalga) ning paar liigest (küünarnukke ei ole, ainult põlved), kuid jala- ja käelabased me talle ei lisa.

Taageldamine (*Rigging*)

Taageldamiseks pead:



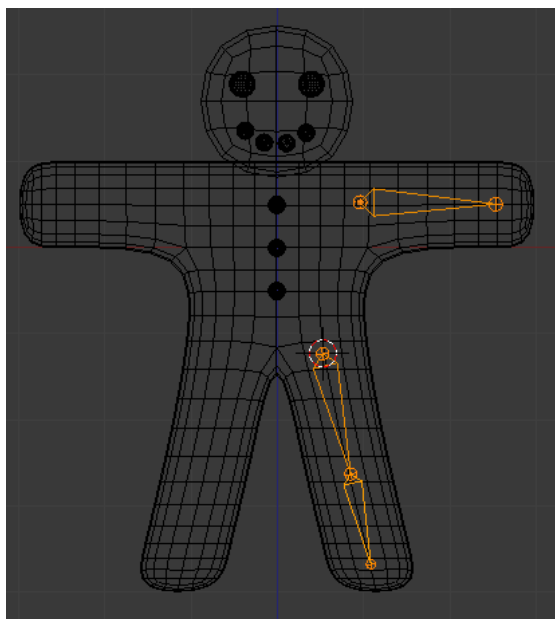
Esimese luu lisamine, küünarnukita käsi.

- Objektirežiimis Object aseta oma 3D-kursor Gusi öla kojale ning vajuta ja vali \diamond ShiftA >> Add >> Armature >> Single bone. Ilmub rombikujuline objekt, mis tähistab skeletisüsteemi luud. Sisene muutmisrežiimi Edit. Luu lõpp ehk ots on valitud (kollane).
- Muutmisrežiimis Edit haara luus otsast kinni (G) ja liiguta see Gusi käelaba kohale - pilt (*Esimese luu lisamine, küünarnukita käsi*). Praegu me teisi luid ei vaja. Nüüd peaks sul olema üks luu, mis jookseb ölast käelabani. Kui sa otsa liigutad, paned tähele, et terve luu muutub suuremaks - sa lihtsalt tegeled luu mõõtkava muutmisega.



Et see protsess edukam oleks, vaata palun aegajalt Piparkoogimeest ja skeletti erinevatest vaatenurkadest, et olla kindel, et luustik asetseb piparkoogimehe sees samamoodi nagu luud asetsevad inimkeha sees. Katmine ei õnnestu, kui luud asuvad näiteks keha ees või taga. Erinevatest vaatenurkadest kontrollimine on tavapärane 3D-mudeli loomise tehnika.

Luut otstest

Luu otstel võivad olla erinevad nimed. Blenderis kutsutakse neid praegu "pea"/"saba" (*head/tail*), esimene tähistab "suurt" otsa, teine "peenikest" otsa). Kuid ajalooliselt on neid nimetatud ka "juur"/"tipp" (*root/tip*), mida peetakse sageli natuke vähem segaduskitavaks...



Teise luu lisamine, jaluude ahel.

- Jää muutmisrežiimi Edit, aseta kursor puusaliigese kohale ning lisa uus luu ⇧ ShiftA.
- Haara (G) uue luu kollasest otsast kinni ja liiguta see põlve kohale.
- Nüüd "lisa ahelana" (*chain*) uus luu, mis läheb põlvest jalalabani, vajutades kombinatsiooniga Ctrl LMB  jalalaba peale. Automaatselt ilmub uus *aheldatud* luu, mis on lingitud põlvega ja lõppeb jalalabas - pilt (*Teise luu lisamine, jalaluude ahel*). Teine võimalus uue luu ahelana lisamiseks oleks lihtsalt selle eendamine kiirvalikuga E. See võimalus tekitab uue luu ja saadab sind automaatselt haaramise/liigutamise režiimi. Vajuta klahvile LMB , et parajasti valitud luu tipu asukoht kinnitada.

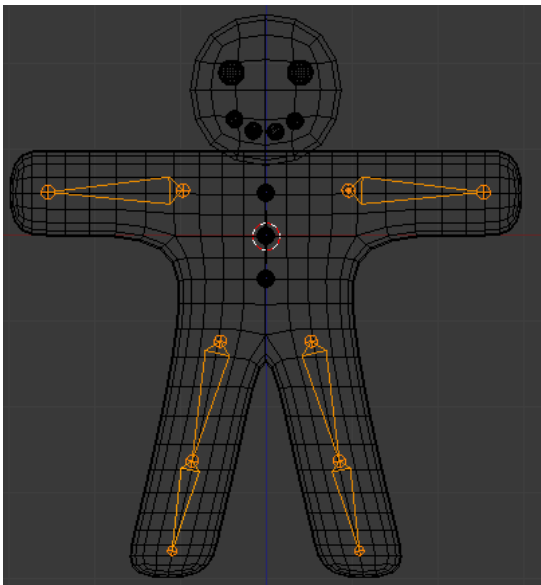
Luu asukoht

Luud, mida me lisame, moonutavad Gusi kehavõret. Et tulemus oleks kena, proovi asetada luude liigeseid piltidel näidatud kohtadesse.


Luu keere

Et luud reastuksid nii nagu näidatud pildil (*Teise luu lisamine, jalaluude ahel*) võib sul tekkida vajadus luude keeret (Bone Roll) muuta vajutades peale alumise jala luu valimist CtrlN, 3.

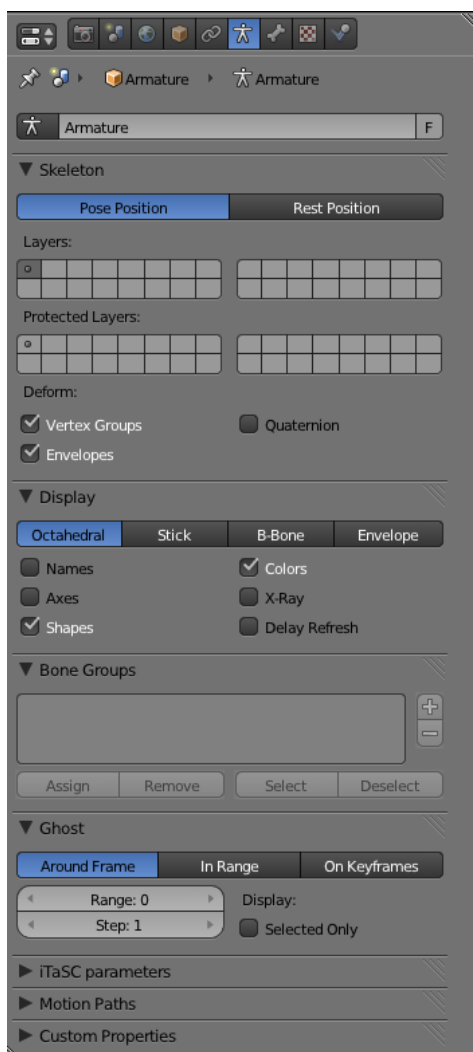
Nüüd on meil kolm luud, mis moodustavad Gusi skeleti.



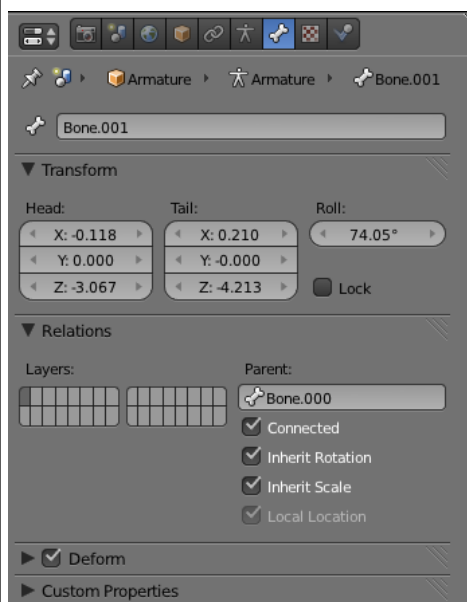
Terve sekelett peale duplitseerimist ja ümber pööramist.

- Aseta nüüd kursor keskkoha (⇧ ShiftC) ja vali kõik luud (A). Duplitseeri nad kiirvalikuga ⇧ ShiftD ja välju koheselt haaramise režiimist kiirvalikuga Esc. Tee kindlaks, et pööramise/möötkava teisendamise teljeks on kursor (pöörlemiselje rippmenüü Pivot 3D-akna päises). Pööra duplitseeritud luud mööda X-telge sõltuvalt kursori asukohast tagurpidi (CtrlM ja seejärel X). Vajuta klahvi LMB , et peegeldamine kinnitada. Tulemuseks on pilt (*Terve sekelett peale duplitseerimist ja ümber pööramist*).

Kui valitud on mõni luu, näitab objekti andmete kontekst Object Data terve skeleti sätteid nagu skeleti objekti nimi ja selle kuvamise võimalused, samas kui luu kontekst Bone näitab aktiivse luu (Bone) nime ja luu enda septsiiifilisi sätteid.



Skeleti kontekst.



Skeleti luu kontekst.

Pane linnuke nimede kasti Names (skeleti kontekst Armature, kuvamise paneel Display), et nimed oleksid 3D-vaates nähtavad, vali seejärel klahviga **LMB** iga luu eraldi, ning vajuta luude kontekstis Bones luu nime peale, et see millekski sobilikuks muuta - näiteks *Kasi.R*, *Kasi.L*, *YleJalg.R*, *AlaJalg.R*, *YleJalg.L* and *AlaJalg.L*, vaata pilti (*Skeleti luu kontekst*). Muutmisrežiimist Edit välju klahviga **↵** Tab.

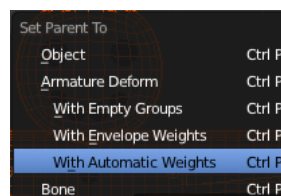
Luude nimetamine

Väga oluline on luu nimede lõppu panna ".L" või ".R" (vastavalt *Left* ehk vasak ja *Right* ehk parem), et tegevuse redaktor Action Editor suudaks automaatselt poose tagurpidi pöörata.

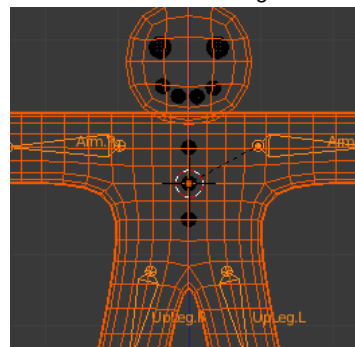
Katmine

Nüüd peame saavutama sellise tulemuse, et skeleti moonutus tekitaks vastava keha moonutuse. Seda teeme me *katmisega* (*skinning*), mis määrab luudele tipud, nii et viimased sõltuksid esimeste liikumisest.

- Objektirežiimis Object vali Gusi keha ning seejärel klahvi **⇧** Shift all hoides ka skelett, nii et keha oleks tumeoranž ja skelett oleks heleoranž.
- Nüüd peame me keha skeletile allutama (parent). Selle saavutamiseks vajuta **CtrlP**. Ilmub (*Allutamise menüü Parenting*). Vali sissekannne Armature Deform >> With Automatic Weights.



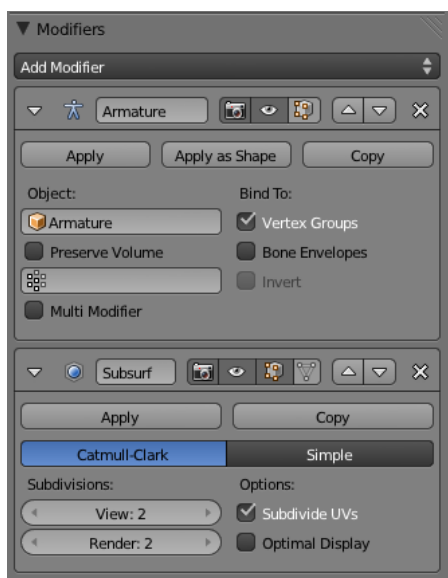
Allutamise menüü Parenting.



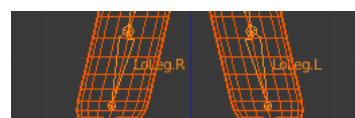
Tipugrupid, ümbrikud ja töötlejate järjekord

Võre katmisel lisatakse võre objekti töötleja nimekirja lõppu skeleti töötleja. Gusi võrel juba on nimekirjas pinnatükeldamise töötleja (*Subdivision Surface*). Et võre moonutamise tulemus oleks sujuv, tuleks skeleti töötleja liigutada pinnatükeldamise töötlejast nimekirjas ettepoole. Selleks vajuta töötleja kontekstis *Modifier* töötleja liigutamise nuppudele *Move Modifier* - pilt (*Töötlejate nimekirja töötlejate kontekstis Modifiers*). Samuti on igal luul mõjupiirkond, mida kutsutakse ümbrikuks (*Envelope*). Skelett moonutab võret nii määratud tipugruppide kui luu ümbrike kohal. See võib tekitada soovimatuid tulemusi ning seetõttu tuleks antud juhul skeleti töötlejas (*Armature Modifier*) luu ümbrikud (*Bone Envelopes*) välja lülitada - pilt (*Töötlejate nimekirja töötlejate kontekstis Modifiers*).

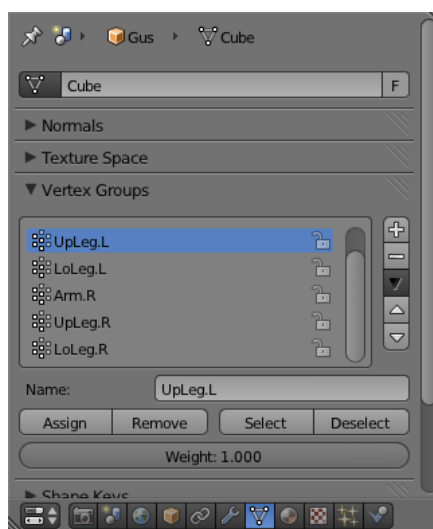
- Vali nüüd Gusi keha ja mine töötlemisrežiimi Edit (**↵** Tab). Pane tähele, et objekti andmete kontekstis Object Data on tipugruppide paneelis Vertex Groups olemas tipugruppide sätted - pilt (*Tipugruppide sätted objekti andmete kontekstis Object Data*).



Töötlemate nimekirja töötlemate kontekstis Modifiers.

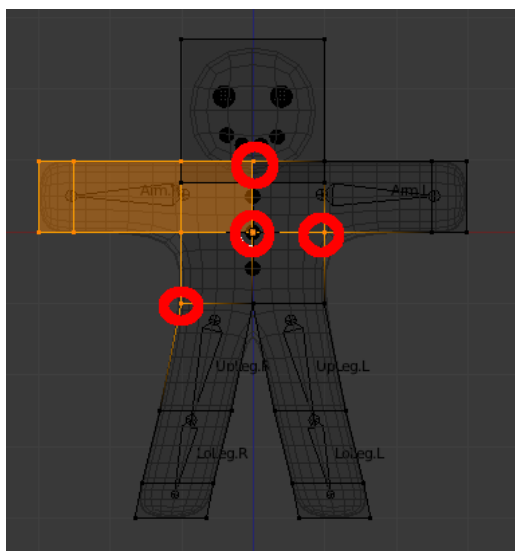


Allutatud skelett.



Tipugruppide sätted objekti andmete kontekstis Object Data.

Tipugruppide paneeli kerides näed kõiki kuute olemasolevat tipugruppi - antud juhul on neid kuus. Aga tõeliselt keeruliste tegelaste puhul, kelle käed ja jalad on täielikult taageldatud, võib neid grupe olla kümneid! Vaata pilti (*Tipugruppide sätted objekti andmete kontekstis Object Data*). Valimise ja valiku tühistamise nuppudega Select ja Deselect saad kõiki grupi tippe valida või seda valikut tühistada ning seeläbi näha millised tipud kuuluvad millisesse gruppi.



Gus muutmisrežiimis Edit koos kõigi gruppi *Kasi.R* valitud tippudega.



Vali parema käe grupp (*Kasi.R*) ja, juhul kui kõik selle käe tipud on valimata (vajadusel vajuta A), vajuta valimise nuppu Select. Tulemuseks peaks olema midagi pildi (*Gus muutmisrežiimis Edit koos kõigi gruppi *Kasi.R* valitud tippudega*) samast.

Kui sa ei näe sama pilti, siis panid sa eeldatavasti luud täpselt õigesse kohta ning *automaatsele katmisel* õnnestus tipud paremini

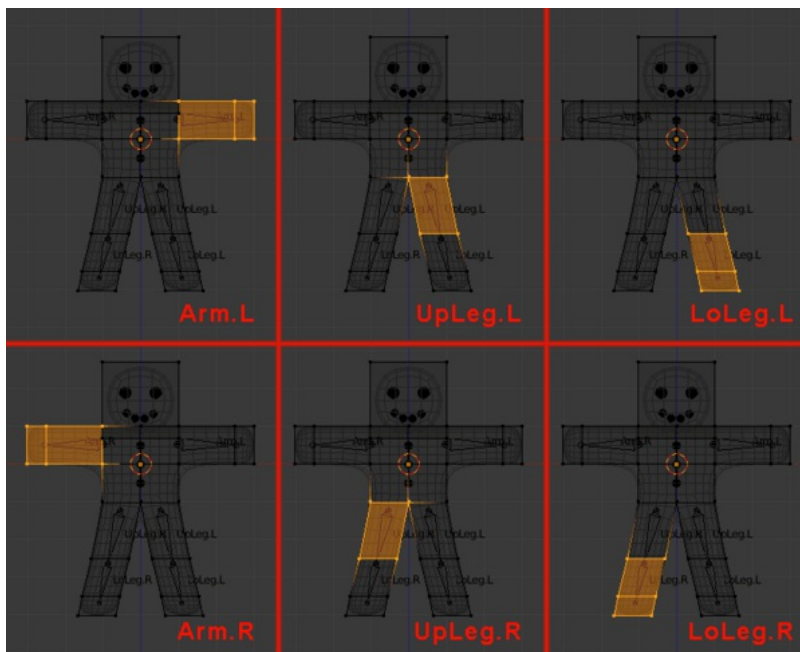
luudega sobitada. See on väga ebatõenäoline, et katmisel sobitatakse luud tippudega nii täpselt nagu sa oodata võiksid. See nõuab, et sa hakkaksid grupeerimist käsitsi täpsustama, nagu seda seletatakse järgmistes osades.

Pildil (*Gus muutmisrežimis Edit koos kõigi gruppi Kasi.R valitud tippudega*) punaste ringidega tähistatud tipud kuuluvad moonutusgruppi, kuigi nad tegelikult ei tohiks selles olla.

Automaatne katmine leidis, et nad asusid luule väga lähedal ning liitis nad seetõttu moonutusgrupiga. Meie neid sellesse gruppi ei taha, sest mõningad asuvad Gusi vastasküljel ja mõned tema rinnas ning nende lisamisel moonutatakse ka noid kehaosaid.

Et neid grupist eemalda, tühistada kõigi teiste tippude valik (nende, mis peaksid gruppi *jääma*) kasutades piirdkastiga valimist (B), kuid luues kasti klahviga MMB , mitte LMB , et kõigi sellesse kasti jäävate tippude valik tühistatakse.

Kui valituks on jäänud ainult "soovimatu" tipud, vajuta eemaldamise nuppu Remove (pilt (*Tipugruppide sätteid objekti andmete kontekstis Object Data*)), et need grupist Kasi.R eemaldada. Tühistada kõigi tippude valik (A) ja kontrolli siis järgmist gruppi. Kontrolli nad kindlasti üle, et tulemus näeks välja nagu pildil (*Kuus tipugruppi*).



Kuus tipugruppi.

Tipugruppid

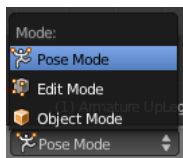
Ole tippude gruppidesse määramisega või nende sealt eemaldamisega väga ettevaatlik. Kui sa näed hiljem ootamatuid moonutusi, siis võisid mõned tipud unustada või lisada gruppi liiga palju tippe. Tipugruppe saad iga kell muuta.

Muud detailid

Meie moonutused mõjutavad ainult Gusi keha ning mitte tema silmi, suud või nuppe, mis on eraldi objektid. Kuigi me selle probleemiga antud lihtsa animatsiooni puhul ei tegele, tuleb seda keerulisemate projektide puhul arvesse võtta - näiteks allutades (tippudele) erinevaid kehaosasi või ühendades neid mingil muul moel üheks võreks (kõiki sätteid on kirjeldatud [manuaalis](#)).

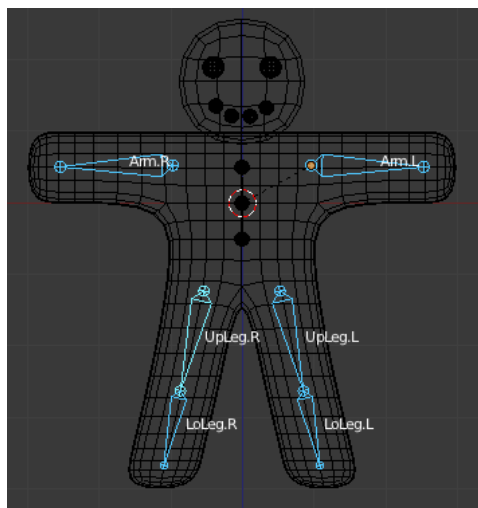
Poseerimine

Kui sa oled Gusi ära taageldanud ja katnud, saad temaga nagu nukkuga mängima hakata, liigutades tema luusid ja vaadates tulemust.



Režiimi menüü 3D-akna päises.

- Vali ainult skelett ning seejärel vali režiimide menüüst Mode poseerimisrežiimi sissekanne Pose Mode - pilt (*Režiimi menüü 3D-akna päises*) – või vajuta lihtsalt Ctrl+Tab. See võimalus on kasutatav ainult siis, kui valitud on skelett.
- Skeleti valitud luud muutuvad sinisteks. Nüüd oled sa poseerimisrežiimis Pose. Kui sa luu valid ning seda liigutad (G) või pöörad (R), moonutub keha vastavalt!



Nüüd oled sa poseerimisrežimis Pose!

Algne asukoht

Blender peab luude algse asukoha meeles. Saad skeleti algse poosi taastada, vajutades AltR, et tühistada luude pöörde muutus, ja AltG, et tühistada nende asukoha muutus. Teise võimalusena võid kasutada objektiandmete konteksti Object Data algsuveni nuppu Rest Position, mis näitab sulle ajutiselt algset positsiooni.

Pöördkinemaatika

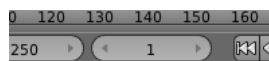
Pöördkinemaatika (Inverse Kinematics, IK) tähistab seda, kui sa määrad luude keti viimase luu asukoha, mille puhul kasutatakse ka nimetust "viimane mõjutaja" ("end effector"). Kõik ülejäänud luud võtavad algorütmil põhineva asukoha, mille arvutab välja *pöördkinemaatika lahendaja (IK solver)*, et ketist ei oleks auke (st pöördkinemaatika lahendab ahela asukohad meie jaoks ise). Selle tulemusena saame me pöördkinemaatika abil käsi ja jalgu palju täpsemalt asetada.

Edaspidine kinemaatika

Poseerimisrežimis Pose luusid liigutades paned sa tähele, et need käituvad kui jäigad, pikendamatud kehad, mille otstes on kerakujulised liigesed. Ahelast saad haarata ainult esimese luu ning kõik teised järgnevad sellele. Kõiki teisi ahela luusid haarata ja liigutada ei saa. Neid saab ainult pöörata nii, et valitud luu pöörduv vastavalt talle eelneva ahela luu pöördele ja kõik järgnevad luud järgivad sama pöördumist.

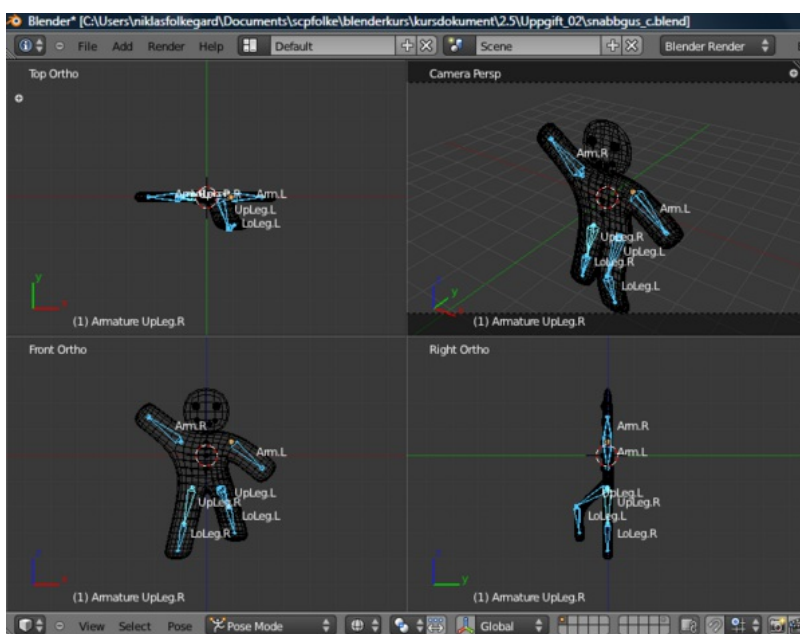
Seda protseduuri, mida kutsutakse *edaspidiseks kinemaatikaks (Forward Kinematics, FK)*, on lihtne mõista, kuid tänu sellele muutub viimase ahela luu asukoha täpne määramine keeruliseks.

Paneme Gusi edaspidise kinemaatika abil jalutama, määrares neli erinevat poosi, mis vastavad neljale sammule. Blender teeb ise ära töö, mille tulemuseks on sujuv animatsioon.

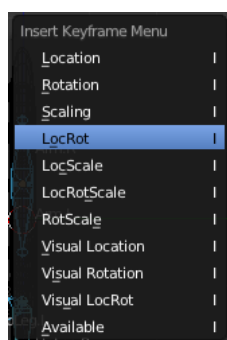


Praeguse kaadri numbriväli ajatelje Timeline akna päises.

- Tee esmalt kindlaks, et oled ajatelje kaadris number 1. Kaadri numbrit näidatakse ajatelje Timeline akna päises olevas numbriväljas - pilt (*Praeguse kaadri numbriväli ajatelje Timeline akna päises*). Kui selle väärtus ei ole 1, määra selleks nüüd 1.
- Pöörates nüüd ühe luu haaval (R), tõstame me luud YleJalg.L ja painutame luud AlaJalg.L tahapoole, tõstes samal ajal luud Kasi.R ja liigutades luud Kasi.L natuke allapoole - pilt (*Meie esimene poos*).

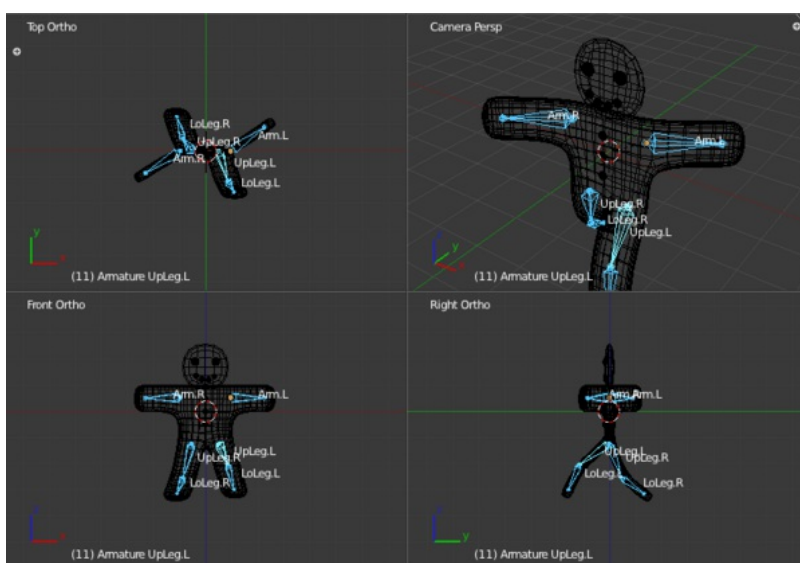


Meie esimene poos.

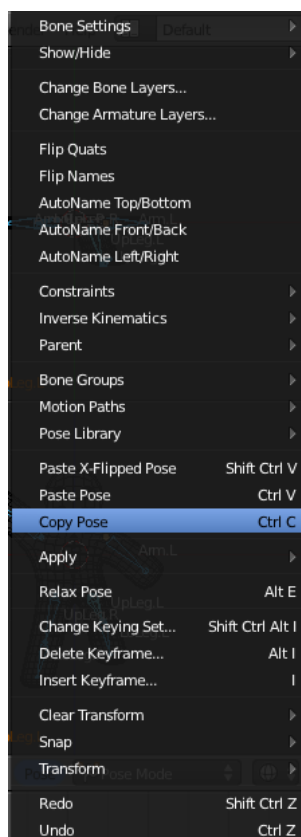


Poosi kaadrisse
salvestamine.

- Vali klahviga A kõik luud. Hoides hiire kursorit 3D-akna kohal vajuta I. Lahti hüppab menüü - pilt (*Poosi kaadrisse salvestamine*). Vali sellest menüüst asukoha ja pöörde sissekanne LocRot. See võtab kõigi luude asukohad ja pöörded ning salvestab need kaadris 1 poosina. See poos kujutab Gusi tema sammu keskel, nii et ta vasak jalg on liikunud ette ja on maapinnast kõrgemal.
- Nüüd mine kaadrisse 11, kas sisestades numbrivälja vastava väärtuse või vajutades ↑. Seejärel liiguta Gus teise asendisse nagu pildil (*Meie teine poos*). Tühista kõige pealt ülal mainitud kiirvalikuga AltR mõlema kae pööre. Ülalvaates olles, pööra luud Kasi.R natuke ettepoole ja luud Kasi.L natuke tahapoole. Lõpeta poos niimoodi, et tema vasak jalg oleks eespool ja parem jalg tagapool ning mõlemad oleksid natuke painutatud. Gus kõnnib ühe koha peal!



Meie teine poos.



Poosi menüü Pose.

- Vali jälle kõik luud ja vajuta I, et poos kaadrisse **11** salvestada ning vali menüüst Rot.
- Nüüd on meil vaja kolmandat poosi kaadrisse **21**, kus parem jalg on üleval, sest me oleme sammu teise poole keskel. See poos on peegeldatud versioon poosist, mille me määrasime kaadrisse **1**. Seetõttu mine kaadrisse **1** tagasi, vali kõik luud ning seejärel 3D-vaate päises olevast poosi menüüst Pose poosi kopeerimise sissekanne Copy Pose - pilt (*Poosi menüü Pose*). Või kasuta kiirvalikut CtrlC. Nüüd oled sa praeguse poosi puhvrissse kopeerinud.
- Mine kaadrisse **21** ning kleebi see poos poosi menüü Pose X-teljel peegeldatud poosi kleepimise sissekandega Paste X-Flipped Pose - pilt (*Poosi menüü Pose*). Või kasuta kiirvalikut ⇧ ShiftCtrlV. See sissekanne peegeldab poosi vahetades järelliideet “.L” omavate luude asukoha järelliidet “.R” omavate asukohaga ehk siis poos peegeldatakse!

Luu keere

Kui X-teljel peegeldatud poosi kleepimisel paadub Gus valet pidi, võib sul olla mõningaid probleeme luu keeretega (*Bone Roll*). Vali muutmisrežiimis Edit kõik luud ja vajuta CtrlN, 3, et luude keerded korda saada. Seejärel mine kaadritesse 1 ja 11 ning paranda seal olevaid poose. Kopeeri kaader 1 uuesti ning kleebi see X-teljel peegeldatuna kaadrisse 21.

Poos on olemas, kuid seda pole veel salvestatud! Pead valima kõik luud ja vajutama I » Rot.

- Nüüd tee sama protseduur kaadri **11** poosiga, kleepides selle peegelpildid kaadrisse **31**.
- Et tsükkel lõpuni viia, kopeeri kaadri **1** poos ja kleebi see *ilma* peegeldamata kaadrisse **41**. Selle saavutamiseks kopeeri poos nagu tavaliselt ja kleebi ta menüü poosi kleepimise sissekandega Paste Pose. Või kasuta kiirvalikut CtrlV. Lõpeta jada salvestades poos kiirvalikuga I » Rot.

Animatsiooni kontrollimine

Oma animatsiooni eelvaatamiseks mine kaadrisse **1** ja vajuta 3D-aknas klahvikombinatsiooni AltA.

Gus kõnnib!

See üksik kohapeal toimuv samm on kõndimise tuumaks ning kui sa oled selle ära defineerinud, on olemas erinevad tehnikad, et tegelast mööda keerulist rada jalutama panna. Kuid meie kiire õpetüki jaoks sellest ühest kohapealsest sammust piisab.

- Omaduste akna Properties renderdamise kontekstis Render määra alguskaadri (Start) väärtuseks **1** (vaikimisi see peakski juba **1** olema) ja lõpukaadri (End) väärtuseks **40** (see on vaikimisi väärtusega **250**, vaata pilti *Animatsiooni sätted renderdamise kontekstis Render*). Kuna kaader **41** on kaadriga **1** ühesugune, peame me täieliku tsükli tegemiseks renderdama ainult kaadrid **1** kuni **40**.



Animatsiooni sätted renderdamise kontekstis Render.

- Kirjuta väljundi paneelis Output olevasse tekstikasti *//render/*.
- Vali formaadi paneelist Format failitüübiks AVI Raw. Kuigi see ei ole enamasti parim valik (eriti faili suuruse probleemide tõttu), on see kiire ja töötab igal masinal, nii et sobib meie vajadustega. Samuti peaksid sa valima AVI Jpeg, et tulemuseks olev fail oleks kompaktsem. Kuid see valik kasutab kadudega JPEG pakimismeetodit ning tulemuseks on video, mida mõned välised

mängijad ei pruugi suuta mängida.

Lõpuks vajuta renderduspaneelis Render animatsiooni nuppu Animation. Pea meeles, et *kõik* kihid, mida sa soovid animatsioonis kasutada, peavad olema nähtavad! Sinu puhul on nendeks kihid 1 kuni 10.

Renderdamise peatamine

Kui teed vea (näiteks unustad kihi 10 sisse lülitada), siis saad renderdamise protsessi klahviga Esc peatada.

Meie stseen on suhteliselt lihtne ning Blender renderdab iga üksiku pildi neljakümnest vaid paari sekundiga. Vaata, kuidas nad nähtavale tulevad.

Kaadrid

Loomulikult võid sa iga animatsiooni kaadri eraldi pildina renderdada, minnes soovitud kaadrisse ja vajutades renderdamise nuppu RENDER.

Kui renderdamine on lõppenud, peaks sul praeguse kataloogi (see, kus asub su `.blend` fail) alamkataloogis `render` olema fail nimega `0001_0040.avi`. Kataloogi saab muuta väljundi paneelist Output.

Seda faili saad otse Blenderis mängida valides ülemisest menüüst renderdatud animatsiooni mahamängimise sissekande Play Rendered Animation (või kiirvalikuga `CtrlF11`). Animatsioon on automaatselt tsüklikline. Mängimise peatamiseks vajuta Esc. Me oleme tekitanud väga algelise kõndimistsükli. Blenderis on veel palju võimalusi, mida sa [manuaali](#) lugedes ka varsti teada saad!

Failide avamine

Mode: Kõik režiimid

Hotkey: F1

Menu: File » Open

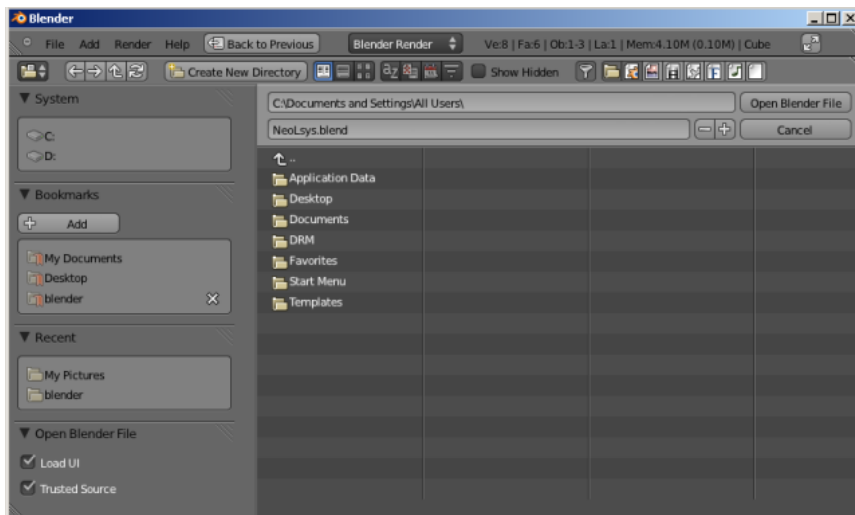
Kirjeldus

Blender kasutab `.blend`-faili vormingut peaaegu kõige salvestamiseks: objektid, stseenid, tekstuurid ja isegi kõik sinu kasutajaliidese sätted.



Blender eeldab, et sa tead, mida teed! Kui avad faili, siis **ei küsita** sinu käest, kas soovid lahtioleva stseeni salvestamata muudatusi salvestada, sest faili avamise dialoogi kinnitamist peetakse piisavaks tõendiks, et sa pole seda valikut teinud juhuslikult.

Ole kindel, et su failid on eelnevalt salvestatud.



Failisirvija kasutamine ja kataloogides navigeerimine

Kettalt mõne Blenderi faili avamiseks vajuta F1. Avaneb eespool näidatud aken File Browser.



Ülemine tekstikast näitab praeguse kataloogi asukohta ja alumine sisaldab valitud faili nime. P (või nupp P) viivad su ühe kataloogi võrra kõrgemale.

+ ja - nuppudega saad liikuda läbi nummerdatud failide: kas suurendades või vähendades faili nime lõpus olevat numbrit.

Klõpsa sisenemiseks kataloogi nimele või klõpsa avamiseks faili nimel ja vajuta seejärel faili avamise nuppu (Open Blender File).

Katkestamisnupu Cancel vajutamine suleb failisirvija akna ja näitab eelmisena avatud olnud akent.

Kõrvalpaneel

Vasakul olev paneel näitab erinevaid faili leidmise viise ja seadistusvõimalusi. Faili laadimiseks vali see LMB  ja vajuta ↵ Enter või klõpsa nupul Open File. Faili saab ka lihtsalt avada, vajutades selle nimel MMB .

Süsteem (System)

Süsteemi menüüs on loetelu ketastest, millele on võimalik failide leidmiseks navigeerida. Vajuta neist ühele, et ketas avada.

Järjehoidjad (Bookmarks)

Järjehoidjad on kataloogid, mida soovid tihti kasutada ja mida sa ei soovi failisirvijas kaua otsida. Kataloogi järjehoidjaks muutmiseks leia see üles ja vajuta lisamise nuppu (Add). Kataloogi eemaldamiseks järjehoidjate hulgast vajuta selle kõrval olevat X ikooni.

Hiljutised (Recent)

See on hiljuti külastatud kataloogide nimekirj. Sul on võimalik valida, kui palju katalooge see nimekirj sisaldab, minnes menüüsakile File kasutajaeelistustes (Preferences) ja muutes kasti Recent Files sisu.

Avamise valikud

Blender salvestab igasse `.blend`-faili kasutajaliidese, teisisõnu ekraanipaigutuse. Vaikimisi see kasutajaliides ka laaditakse ning muudetakse ära kõik kasutaja eelistused ja kehtiv ekraanipaigutus. Kui soovid töötada `.blend`-failiga nii, et su praegused eelistused säiliks, käivita uus Blenderi aken ja ava failisirvija (F1). Tühista nupu Load UI valik ja ava fail.

Päise paneel

Päises asuvad erinevad tööriistad failides navigeerimiseks. Neli nooleikooni lubavad sul:

- Move to previous folder - Liikuda eelmisse kataloogi
- Move to next folder - Liikuda järgmisse kataloogi
- Move up to parent directory - Liikuda ülemisse kataloogi
- Refresh current folder - Värskendada praegust kataloogi

Praegusse kataloogi uue kataloogi loomiseks vajuta ikooni Create New Directory.

Teised ikoonid aitavad sul valida, milliseid faile ja kuidas kuvatakse. Võid:

- Display files as a short list - Näidata faile lühikese nimekirjana
- Display files as a detailed list - Näidata faile detaile esitava nimekirjana
- Display files as thumbnails - Näidata faile eelvaatepiltidena

Faile saad sorteerida:

- Alphabetically - Tähestiku järjekorras
- By file type - Failitüübi järgi
- By Date of last edit - Viimase muutmise aja järgi
- By file size - Faili suuruse järgi

Filtreerimine valib seda, milliseid failitüüpe näidatakse. Filtreerimiseks vajuta Enable Filtering ikooni ja vali, milliseid tüüpe näidatakse:

- Folders - Kataloogid
- Blend files - .blend-failid
- Images - Pildid
- Movie files - Videofailid
- Scripts - Skriptid
- Font files - Fondifailid
- Music files - Helifailid
- Text files - Tekstifailid

Muud failiavamise võimalused

Menüüst File saad avada faile ka järgnevate tööriistade abil:

Ava hiljutisi (Open Recent)

Loendab viimati avatud faile. Laadimiseks klõpsa nimel.

Laadi viimane sessioon (Recover Last Session)

See valik laadib `quit.blend` faili, mille Blender salvestab automaatselt hetk enne sulgumist. Nii saad avada oma viimase töö näiteks juhul, kui sulgesid Blenderi kogemata...

Taasta automaatsalvestusest (Recover Auto Save)

See valik avab kättesaamiseks viimase automaatselt salvestatud faili.

Failide salvestamine

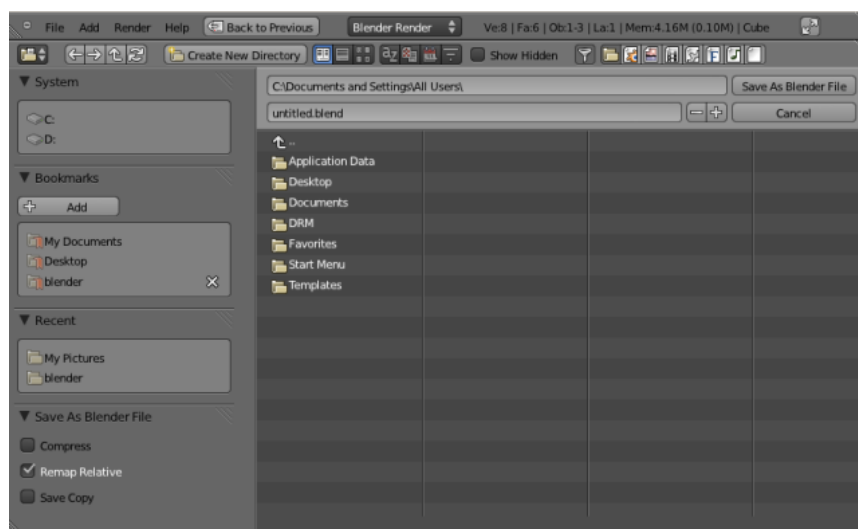
Mode: Kõik režiimid

Hotkey: F2

Menu: File » Save

Kirjeldus

Failide salvestamine on sarnane nende laadimisega. Vajutades F2, avaneb failisirvija aken (File Browser). Ilmunud aken on samasugune nagu failide avamise oma, välja arvatud mõned valikud kõrvalpaneelil. Failisirvija kirjelduse ja kasutamise kohta vaata [failide avamise](#) lehelt.



Salvestamine

Faili nime sisestamiseks vajuta alumisele tekstikastile. Kui nimetus ei lõppe laiendiga “.blend”, lisatakse laiend automaatselt. Seejärel vajuta ↵ Enter või klõpsa salvestamispul Save File.

Kui samanimeline fail juba eksisteerib, pead selle ülekirjutamise kinnitama Save Over dialoogis.

Olenevalt sellest, kui palju [salvestamisversioone](#) oled seadistanud, nimetatakse kõigi samanimeliste failide laiend jooksvalt ümber laiendiks .blendn, kus *n* on 1, 2, 3 jne. Kui töötasid failiga MinuProjekt.blend ja salvestasid selle, nimetati olemasolev MinuProjekt.blend ümber MinuProjekt.blend1 ning salvestati uus MinuProjekt.blend. Nii jäävad sulle vanadest salvestustest arhiveeritud versioonid, mida saad avada, kui pead suures koguses muudatusi ennistama.

Salvestamise valikud

Salvestamise seaded asuvad külgsuuna alumises otsas.

Failide pakkimine (*Compress Files*)

Kui suurte failide mahtu vähendada soovid, muuda pakkimise valik (File » Compress File) aktiivseks.

Säilita sõltuvused (*Remap Relative*)

See valik kaardistab uude kohta salvestamisel kasutusel olevad failid ja korrigeerib vajadusel nende katalooge.

Salvesta koopia (*Save Copy*)

See salvestab hetkeseisust koopia, kuid ei muuda seda koopiat koheselt Blenderis aktiivseks.

Nõuanne failide nummerdamiseks

Salvestamise dialoog sisaldab lihtsat võimalust, kuidas oma failidest erinevaid versioone teha: + NumPad või - NumPad suurendab või vähendab faili nime lõpus olevat numbrit. Kui soovid lihtsalt avatud faili üle salvestada ja salvestamise dialoogi vahele jätta, vajuta F2 asemel Ctrl+W ja kinnita oma valik klõpsuga.

COLLADA import ja eksport



This page has been marked for development documentation changes.
Feel free to comment on the [talk page](#).

WIP REASON:

Töös olev dokumentatsioon, omadus on ikka veel arendamisel, [vaata siit](#)

Võre

Import

Lubatud geomeetria tüübid on

- kolmnurgad (*tris*)
- hulknurkade loendid (*polylist*)
- hulknurgad
 - n-gonid muudetakse automaatselt kolmnurkadeks
- kolmnurkade lehvikud

Eksport

Võre andmed eksporditakse hulknurkade loendi märgisega <polylist>.

Valgus

Import

Blender annab endast parima, et valguseid .dae-failist importida. Kui valguste kohta leitakse Blenderi profiil, kasutatakse sellest leitud väärtusi. See kindlustab, et kõik Blenderist eksporditud .dae-failid imporditakse hiljem sajaprotsendiliselt. Blenderi versioonis 2.57 lisati märgise <extra> tugi.

Eksport

Blenderi profiil valguste jaoks on lisatud märgise <extra> abil. Selle profiili abil eksporditakse Blenderist kogu Valguste andmestruktuur peale valguse hajususkõvera.

Materjal & efektid

Efektide eksportimisse on alates Blenderi versioonist 2.57 tehtud mõningaid muudatusi. Neist olulisem on see, et <lambert> eksporditakse ainult siis, kui läike väärtus on 0.

Sõlmed

Importimisel antakse <instance_node> sõlme-märgiste ülemteisendused korrektselt edasi olemasolevatele alamsõlmedele. Blenderi materjalid eksporditakse järgnevate laotustega:

- Phong
- Blinn
- Lambert

Olulised asjad, mida meeles pidada

- objektide ja andmeblokkide nimede maksimumpikkus on 21 baiti (tähemärki).
- UV-kihtide nimede maksimumpikkus on 32 baiti (tähemärki).
- skeleti animatsioonid ainult võrel, ühe naha kontrolleriiga
- töötlejate tuge veel ei ole

<instance_node> sõlme-märgistega eksporditud .dae-faili importides see informatsioon põhimõtteliselt kaob ja need sõlmed muutuvad <node> märgisteks.


Ülevaade

Igas .blend-failis on andmebaas. Selles andmebaasis sisalduvad kõik failis leiduvad stseenid, objektid, võred, tekstuurid jne. Failis võib olla mitu stseeni ja igas stseenis mitu objekti. Objektidel võib olla mitu materjali, mis võivad omakorda sisaldada mitmeid tekstuure. Ka on võimalik erinevate objektide vahele linke luua.

Mode: Kõik režiimid, iga aken


Hotkey: ⇧ ShiftF4 - Data Select Browser

Andmebaasi sisenemiseks vajuta ⇧ ShiftF4 ja programmi aken muutub andmete sirvijaks (Data Select Browser), milles on üles loetud sinu .blend-faili objektid. Kui soovid minna kõrgemale astmele, vajuta navigatsiooniribale (..) ja sa näed faili üldist struktuuri: tegevus (Action), skelett (Armature), pintsel (Brush), kaamera (Camera), kõver (Curve), grupp (Group) ja nii edasi (kaasa arvatud objektid (Object)).

LMB  abil ükskõik millise andmebloki tüübi valimine, näiteks võre (Mesh), näitab sulle kõigi failis sisalduvate võrede loendit koos sellega, kui mitu kasutajat neist igale on. Kui sul näiteks on auto võre ja sa kasutad seda võret parkimisplatsi stseenis kuue auto jaoks, näitaks Mesh sulle Autot ning selle järel numbrit 6.

Mode: Andmete sirvija Data Select Browser

Hotkey: F

RMB  abil mõnd tüüpi andmeblokkide valimisel (materjal Material, pilt Image, tekstuur Texture...) ja F vajutamisel määratakse neile andmeblokkidele "võltskasutaja". Võltsitud kasutajaga andmete puhul hoiab Blender andmeblokke isegi siis failis alles, kui neil puudub "tegelik kasutaja". Andmeblokke, millel puudub võltsitud või tegelik kasutaja, .blend-faili ei salvestata. F uuesti vajutamine eemaldab võltsitud kasutaja. See tegevus on sama nagu materjali/pildi/... nime kõrval F nupule vajutamine.

Ülevaade ja OOPS-skeem

Oma faili sisu saad lihtsalt uurida ülevaate akna Outliner kaudu. Selles aknas kasjastub Blenderi andmesüsteem, mis on täielikult dokumenteeritud ([siin](#)). Aken näitab kaht andmebaasivaadet. Ülevaade Outliner lubab sul objektidega lihtsamaid tegevusi läbi viia. Nende tegevuste hulgas on valimine, ümbernimetamine, kustutamine ja linkimine. OOPS-skeemi vaade OOPS Schematic aitab sul lihtsasti näha, kuidas andmeblokid on omavahel lingitud. (OOPS - ObjektOrienteeritud ProgrammeerimisSüsteem) Päises olevate nuppude abil saab vaate sisu filtreerida.

Kasutajad (jagamine)

Paljusid andmeblokke saab teiste andmeblokkidega jagada - taaskasutamine on soovitatav. Kujutleme näiteks, et sul on ühel objektil materjal nimega "Läikiv". Võid valida mõne teise objekti, millel materjali veel määratud pole. Selle asemel, et vajutada materjali lisamiseks ADD NEW, vajuta hoopis uue materjali lisamise nupu kõrval olevat väikest üles-alla noolt, mis muudab nähtavaks juba olemasolevate materjalide nimekirja. Vali "Läikiv". Nüüd jagavad need objektid sama materjali. Sa märkad kindlasti materjali nime kõrval numbrit "2", mis näitab, et sellel materjalil on kaks kasutajat (need kaks objekti). Teisteks tavapärasteks näideteks on:

- Tekstuuride jagamine materjalide vahel.
- Võrede jagamine objektide vahel ("kloonid").
- IPO kõverate jagamine objektide vahel, näiteks selleks, et panna kõiki valgusi korraga hämarduma.

Võltskasutaja

Pea meeles, et Blender ei salvesta andmeblokke, mis pole "praeguses" failis mitte millegi külge ühendatud. Kui ehita ".blend"-faili, mis käituks teegina asjadest, millele kavatsed "teistest" failidest linkida, pead kindlustama selle, et neid praegusest (teek) failist ei kustutataks. Tee seda, määra andmeblokkidele "võltskasutaja", vajutades F nuppu andmebloki nime kõrval. See takistab kasutajate arvu nulli langemist ning seetõttu andmeblokki ei kustutata. (Blender ei pea arvet selle üle, palju teisi faile sinu failile lingib.)

Objektide kopeerimine ja linkimine stseenide vahel

Mõnikord võid soovida objekte stseenide vahel kopeerida või linkida. See on võimalik, valides objektid, mida linkida või kopeerida soovid, ning kasutades seejärel 3D-vaate päises oleva Object-menüü valikuid Make Links ja Make Single User. Kasuta valikut Make Links, et luua linke stseenide vahel. Päris koopia tegemiseks tuleb kõigepealt teha link ning seejärel valida Make Single User, et teha valitud objektist avatud stseeni sõltumatu koopia. Rohkem informatsiooni stseenidega töötamise kohta leiab [siit](#).

Failidevaheline lisamine või linkimine

Ühe .blend-faili sisule pääseb lihtsalt ligi ning seda saab avatud failile lisada, kasutades File → Append-funktsiooni (sellele pääseb igal ajal ligi, vajutades ⇧ ShiftF1). Uurimaks täpsemalt, kuidas objekte .blend-failide vahel kopeerida või linkida, [vajuta siia](#).

Proksiobjektid

[Proksiobjektid](#) võimaldavad sul muuta (osa) lingitud andmetest kohalikeks. See võimaldab näiteks animaatoril teha tegelase juhtobjektidest kohalikud "koopiad", ilma et terve taageldus duplitseeruks. See on eriti kasulik tegelase animeerimise puhul, kui soovid, et terve tegelane laetaks välisest teegist, kuid animaatoril oleks samas võimalik töötada pooside ja tegevustega. Teise näitena võid lasta modelleerijal töötada auto kuju (võre) kallal samal ajal, kui teine kunstnik tegeleb selle auto materjalidega. Maalija ei saa auto kuju muuta, kuid saab alustada selle värvide valiku kallal töötamist. Auto kujule tehtud täiendused liidetakse automaatselt maalija proksiobjekti külge.

Andmete kokku ja lahti pakkimine

Blender võimaldab lisada (hõlmata) .blend-failidesse erinevat sordi andmeid, mis tavaliselt oleksid väljaspool .blend-faili salvestatud. Näiteks võib pilttekstuuri, mis on välises .jpg-failis, panna .blend-faili "sisse", kasutades valikut File → External Data → Pack into .blend file. Kui .blend-fail salvestatakse, pannakse koopia sellest .jpg-failist salvestatud faili sisse. Seejärel saab .blend-faili kõikjale kopeerida või e-postiga saata ning pilttekstuur liigub sellega kaasa.

Seda, et pilt on kaasa pakitud, näed selle päises kuvatavast väikesest "jõulupakikesest".

Andmete lahti pakkimine

Kui oled saanud eelkirjeldatud pakitud faili, võid selle File → External Data → Unpack into Files... kaudu lahti pakkida. Sulle pakutakse võimalust taastada algne kataloogistruktuur või paigutada fail asukohta // (kataloogi, kus .blend-fail asub). Kasuta "algset asukohta", kui kavatsed tekstuure muuta, kokku pakkida ja .blend-faile vahetada, sest tagasi saatmisel uuenevad lahtipakkimisel ka algse saatja tekstuuride koopiad.

Stseenide haldamise struktuur

Stseenide haldamine ja teekide lisamine/linkimine põhineb Blenderi [teegil ja andmesüsteemil](#), nii et kasulik on lugeda seda kasutusjuhendi lehekülge kõigepealt, kui sa pole selle süsteemi algtõdedega tutav.

Blenderit saab kasutada nii lihtsa üksikstseeni või pildi loomiseks kui ka terve filmi tootmiseks. Film koosneb enamasti kolmest vaatusest:

1. Sissejuhatus-konflikt.
2. Tõusev pinge.
3. Kulminatsioon-lahendus.

Iga vaatus koosneb mõningatest stseenidest ehk tegevuskohtadest. Iga stseen filmitakse eraldi võtteplatsil, laval või toimumiskohas. Igas võttekohas on rekvisiidid ja taust. Üks stseen koosneb sündmuste jadast, mille jooksul mängivad näitlejad (loodetavasti veenvalt). Iga selline sündmus või võte kestab tavaliselt paar sekundit.

Jooksewõte

Mõnikord kestab võte mitu minutit. Seda kutsutakse "jooksewõtteks" ja see võib kesta terve ühe stseeni. Seda tehnikat on keeruline meisterlikult kasutada, et su vaatajaskond magama ei jääks!

Iga Blenderi fail on organiseeritud ja seadistatud nii, et seda saaks kasutada terve filmi jaoks. Iga .blend-fail võib sisaldada mitut stseeni. Stseen on kihtide kaupa organiseeritud objektide hulk. Loomeprotsessi jooksul kasutad erinevaid [ekraanipaigutusi](#), mis on loodud selleks, et saaksid efektiivselt täita erinevaid ülesandeid: objekte modelleerida ja rekvisiite luua, riietada näitlejaid ja kujundada võttepaika (määrata materjale), lisada tegevusi (animeerida), renderdada videot ja luua valmis filmi. Võid neid paigutusi muuta ja luua täiesti uusi paigutusi vastavalt enda tööstiilile.

Ajatelje planeerimine

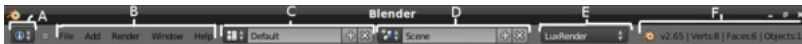
Stseeni siseselt tehakse võtteid, liigutades kaamerat ja/või tegelasi mõne sekundi jooksul läbi stseeni. Blenderis mõõdetakse aega kaadrites, tavalises videos on 25 või 30 kaadrit sekundis (fps) ning filmis 24 kaadrit. Viiesekundilise võtte jaoks tuleb seega plaanida 150 kaadrit (30 fps × 5 sekundit). Andes endale natuke mänguruumi, tuleks teist võtet alustada kaadrist 250. Põhja-Ameerika standardi järgi (NTSC) tehtud üheminutilise filmil, mis on paigutatud ühte stseeni, oleks 1800 kasutuses oleva kaadri pikkune ajatelg, mis võiks siis koos mänguruumiga ulatuda lausa 2500 kaadri. Selline ajatelg lubab sul välja lõigata 700 kaadrit ning valida lõpliku filmi jaoks 1800 kaadrit (30 fps × 60 sekundit = 1800 kaadrit).

Mitu kaamerat

Sul võib ühes stseenis olla erinevate võtete jaoks mitu kaamerat ning sa saad valida, milline neist on iga üksikvõtte renderdamisel aktiivne.

Kehtiv ekraanipaigutus ja stseen

Stseenid pakuvad võimalust oma töö organiseerimiseks. Stseenid võivad omavahel objekte jagada, kuid võivad samas näiteks erineda üksteisest oma renderdamisresolutsioonilt või kaameranurgalt. Hetkel avatud ekraanipaigutust ja stseeni näidatakse kasutaja eelistuste (User Preferences) akna päises, mis tavaliselt asub su ekraani ülaservas:



Kasutajaeelistuste akna päis. **A)** Akna tüübi ikoon, **B)** Menüü, **C)** Ekraanipaigutus, **D)** Stseenid, **E)** Parajasti avatud Blenderi versioon (vajuta vasakul olevale Blenderi ikoonile, et näha tutvustusekraani).

Kasutajaliidese laadimine menüü “File” → “Open” kaudu

Blender salvestab igasse .blend-faili kasutajaliidese ekraanipaigutuse ehk siis selle, kuidas liides faili salvestamise hetkel kohandatud on. Vaikimisi see kasutajaliides ka laaditakse ning muudetakse ära kõik kasutaja eelistused ja kehtiv ekraani paigutus. Kuid sa võid avada .blend-faili nii, et salvestatud liidest ignoreeritakse ja avamisel säilib parajasti seadistatud ekraanipaigutus. Seda saab teha Blenderit taaskäivitades või eelseadistusi uuesti laadides menüü File → New kaudu või vajutades CtrlX ning avades valiku File → Open... või F1 abil failisirvija. Lülita sirvija päises välja nupp Load UI ja ava seejärel fail. Nii ei muuda Blender uut faili laadides su praegust ekraanipaigutust.

Stseenidega töötamine

Vali stseen, millega soovid töötada, vajutades stseeni nime kõrval olevat üles-alla noolt. Stseenid ja nendes sisalduvad objektid on tavaliselt seotud projektiga, mille kallal parajasti töötad. Kuid ka neid saab taaskasutamiseks salvestada, vajutades CtrlU. Nii tehakse ilmuvad nad automaatselt järgmisel Blenderi käivitamisel või kui kasutaja valib File → New (CtrlX).

Blenderi paigalduses on kaasas üks vaikestseen ühe kaamera, valgusti ja kuubiga.

Stseeni lisamine

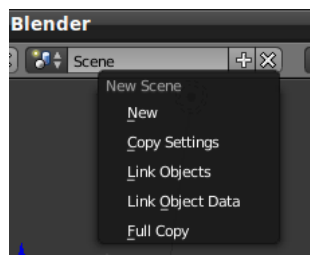
Sa saad teha oma praegusest stseenist täieliku koopia, alustada uuesti tühja stseeniga, või luua stseeni, mis sisaldab linke parajasti avatud stseenile; objektid on küll uues stseenis näha, kuid eksisteerivad tegelikult vanas. Kasuta linkimise võimalust, kui näiteks algses stseenis on tegevuskoht ning uues stseenis peaksid sisalduma tegelased või rekvisiidid.

Uuesti alustamine

Kui alustad uue stseeniga, lisa kindlasti kõigepealt kaamera ja valgustid!

Stseenid on rippmenüüs reastatud tähestiku järjekorras. Kui soovid neid panna teise järjekorda, alusta nende nimetust mõne numbriga, näiteks “1-”. Programm viitab stseenile kolmetähelise lühendiga “SCE”.

Stseeni lisamiseks vajuta stseenide nimekirja nupule ja vali Add New. Stseeni lisamisel on sul järgnevad võimalused:



Stseeni lisamise menüü.

Tühi (Empty)

Tekita täiesti tühi stseen.

Lingi objektid (Link Objects)

Kõik objektid lingitakse uude stseeni. Objektide kihtide ja valikute lipukesed saab iga stseeni jaoks eraldi seadistada.

Lingi objektandmed (Link ObData)


Duplitseerib ainult objektid. Objektidega seotud andmeid (ObData) nagu võre ja kõverad ei duplitseerita.

Täielik koopia (Full Copy)

Kõik duplitseeritakse.

Tavaliselt tehakse esimese stseeni jaoks vaikestseenist täielik koopia. Teise võimalusena võid jääda vaikestseeni ning alustada seal vedeleva kuubi loominguilise muutmisega.

Stseeni nimetamine

Vajutades ⇧ Shift LMB  stseeni nimel (tavaliselt “Scene.001”), saad muuta stseeni nime. Näiteks on “TüdrukKohtabPoissi” tavaliselt esimene kolmest vaatusest.

Seejärel hakkad 2-Model-nimelist ekraanipaigutust kasutades rekvisiite ja objekte modelleerima.

Stseeni linkimine

Igal hetkel saad linkida ükskõik millist objekti ühest stseenist teise. Ava lihtsalt stseen, kus need objektid asuvad, ja vali käsuga CtrlL → To Scene... stseen, kus soovid, et objektid oleksid nähtavad. Nad on lingitud originaalobjektidega; kui soovid neid valitud stseenis muuta üksikkasutaja omadeks (iseseisvateks, linkimatuteks...), vali nad ja vajuta U. Näed mõningaid valikuid, mis lasevad sul soovitud andmeplokke (objekt, materjal, tekstuur...) vabaks anda.

Stseeni eemaldamine failist

Saad parajasti avatud stseeni eemaldada, vajutades selle nime kõrval olevat nuppu X.

Üldvaateaken (*Outliner*)

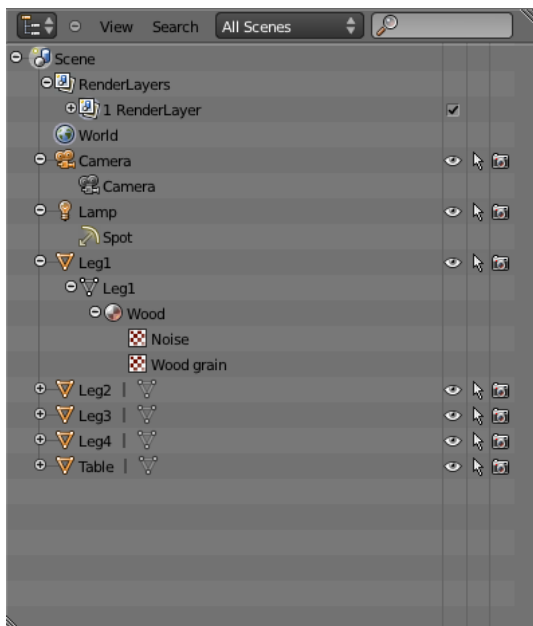
Kirjeldus

Ülevaateakent (*Outliner*) saab kasutada navigeerimise lihtsustamiseks keerulises stseenis. Ülevaade esitab sinu keerulise 3D-maailma kahemõõtmelisena. Kasuta seda stseenist asjade leidmiseks.

Oletame näiteks, et sa aevastad mõnda objekti liigutades (terviseks!), hiir lendab su laualt minema ning objekt heidetakse kuhugi ekraanivälisesse ruumi. Otsi see lihtsalt ülevaatest üles, vali ära ning liiguta 3D-aknasse tagasi, sundides seda kursori külge nakkuma (⇧ ShiftS → Selection → Cursor).

Teine praktilisem näide on mõne muudatuse mõju hindamine sellega seotud [andmeblokkidele](#). Oletame, et sa vaatad oma *LauaPlaat* objekti ning see ei näe õige välja, selle materjal *Puu* pole õige välimusega ja sa soovid, et see näiks rohkem mahagoni moodi. Kuna sama materjali kasutatakse erinevate võrede puhul, pole sa kindel, mitu objekti muudab värvi, kui sa materjali muudad. Ülevaadet kasutades leiad materjali üles ning saad välja selgitada kõik lingid kõigi võredega, mis tal stseenis on.

Ülevaade

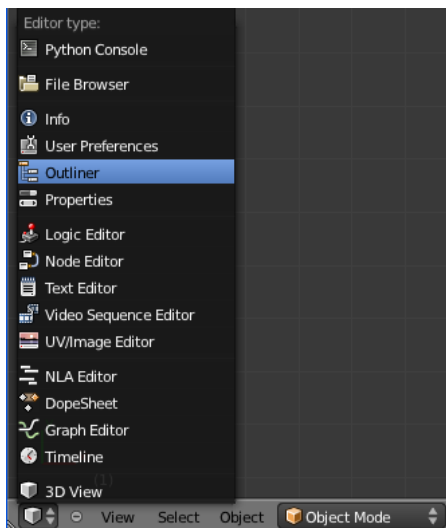


Outlineri aken.



Üldvaade Outliner on teatavat sorti nimekiri, mis organiseerib üksteisega seotud asju. Ülevaates saad:

- Näha stseenis olevaid andmeid.
- Stseenis olevaid objekte valida ja nende valikuid tühistada.
- Muuta stseenis olevaid objekte nähtavaks ja neid peita.
- Valimise võimalust lubada või keelata (et objekt muutuks 3D-vaates "mittevalitavaks").
- Lubada või keelata objekti renderdamist.
- Kustutada stseenis olevaid objekte.
- Andmeid lahti linkida (see on võrdne nupu X vajutamisega andmebloki nime kõrval).
- Lihtsalt valida seda, millist renderduskihti renderdada.
- Lihtsalt valida, millist renderduskäiku renderdada (näiteks võid sa valida, et renderdataks ainult läike (Specular) käik).

Ülevaateakna tüübi valimine



Muuda akna tüüp ülevaateaknaks.



Vali aken ja klõpsa selle aknatüübi nupul (kõige vasakpoolsem ikoon päises) LMB  ning vajuta seejärel LMB  ülevaateakna valikul (Outliner).


Ülevaate kasutamine

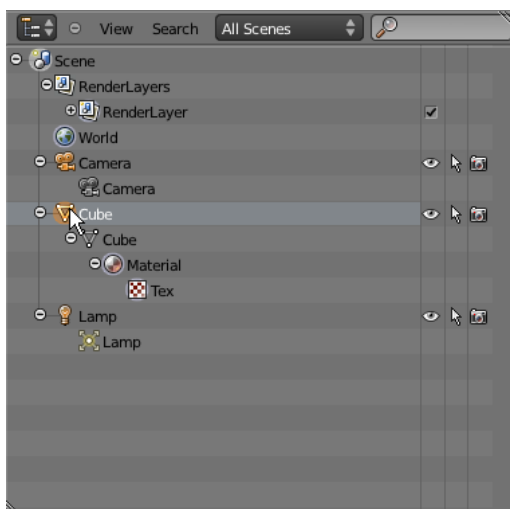
Iga rida ülevaates näitab üht andmeblokki. Võid vajutada parajasti valitud andmebloki nimest vasakule jäävat plussmärki, et näha, milliseid teisi andmeblokke see sisaldab. Kui valitud rida on juba laiendatud, on vasakul olevaks ikooniks miinusmärk. Miinusele vajutamine lükkab rea alamobjektid uuesti kokku.


Ülevaates andmeblokkide valimine ei pruugi veel tähendada seda, et samad andmeblokid valitakse ka stseenis. Stseenis andmebloki valimiseks tuleb see aktiveerida.

Valimine ja aktiveerimine

Üksikute objektide valimine ei nõua eelvalikuid: vali otse nimede/ikoonide ala "sees", kasutades LMB  (ja/või RMB ; avaneb menüü, mida on kirjeldatud allpool).

Kui valid nii nimekirjast mõne objekti, muutub see valituks ja aktiivseks objektiks ka kõigis teistes 3D-vaadetes. Kasuta seda võimalust objektide leidmiseks 3D-vaates: vali nad ülevaate all ning liigu nende juurde, vajutades . NumPad; kui sul numbriklahvistik puudub, pane nad oma kursori külge nakkuma, valides kõigepealt  ShiftS → Cursor → Selection ja vajutades seejärel C.

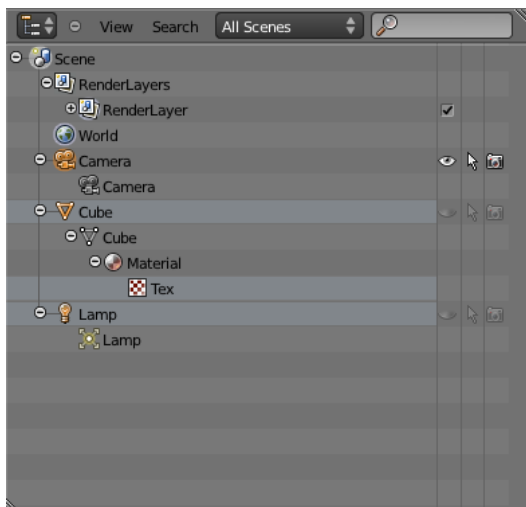


Vajuta kuubi võre andmetel LMB , et aktiveerida muutmisrežiim (Edit mode).

Andmebloki aktiveerimine

Aktiveeri andmeblokk, vajutades LMB  selle *ikoonil*. Andmebloki aktiveerimine lülitab automaatselt sisse vastava režiimi.





Näiteks kuubi võre andmete aktiveerimine valib kuubi ja siseneb muutmisrežiimi, samas kui kuubi objekti andmete aktiveerimine valib kuubi ja siseneb objektirežiimi (vaata joonist paremal).



Andmeblokkide eelvalimine.


Andmeblokkide grupi eelvalimine

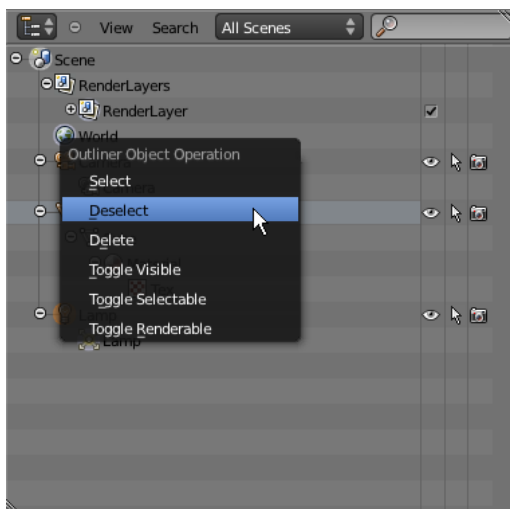
See on kasulik, kui soovid hulgaliselt andmeblokke korruga valida või nende valimist tühistada. Sellise valiku ettevalmistamiseks pead (vastavalt isiklikule eelistusele) kasutama:

- RMB  või LMB ,
-  ShiftRMB  või  ShiftLMB ,

- RMB  ja hiire lohistamist või LMB  ja hiire lohistamist,

ning tegema kõike seda *väljaspool* andmehlokkide nime/ikooni piirkonda. Eelvalitud blokkide read on seejärel märgitud heledama tooniga.

Neid saab seejärel valida ja nende valikut tühistada, vajutades RMB  nime/ikooni piirkonna *peal*, mis käivitab hüpikmenüü (vt altpoolt).



Kuubi objekti hüpikmenüü.

Kontekstipõhine hüpikmenüü

Sisene sellesse menüüsse, vajutades andmehloki nimel või ikoonil RMB . Olenevalt eelnevalt valitud andmehloki või -blokkide tüübist, pakutakse sulle kas kõiki või osasid järgnevatest võimalustest:

- Vali (Select).
- Tühista valik (Deselect).
- Kustuta (Delete).
- Tühista link (Unlink) – et linkida andmehlokk oma “omaniku” küljest lahti (nt materjal seda kasutavast võrest).
- Tee kohalikuks (Make Local) – et luua andmehlokkist “kohalik” koopia.

Märkus: mõnedel andmehloki tüüpidel puudub kontekstipõhine menüü sootuks!

Andmehloki kustutamine

Valitud andmehloki või -blokkide kustutamiseks vajuta X.

Alamandmehlokkide näitamine

Andmehlokkide puus mõne astme sisu avamiseks vajuta + NumPad.

Alamandmehlokkide peitmine

Andmehlokkide puus mõne astme sisu peitmiseks vajuta - NumPad.

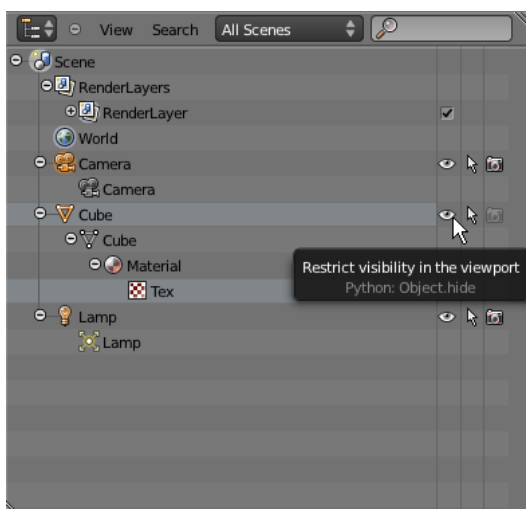
Kõigi astete näitamine/peitmine

Andmehlokkide puu kõigi astete sisu näitamiseks/peitmiseks vajuta A.

Objekti tasemel keeldude muutmine

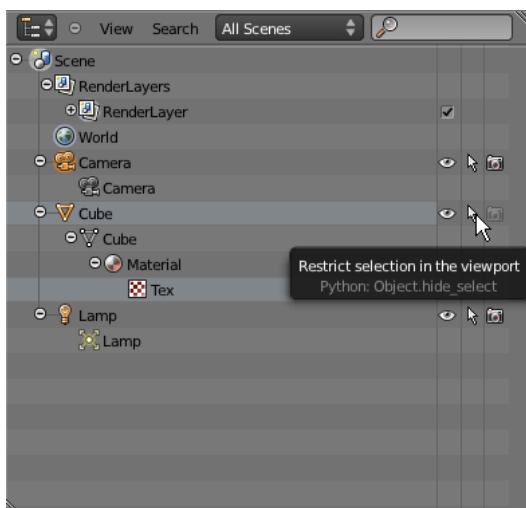
Objektide puhul on ülevaateakna paremas servas järgmised kolm valikut:

Nähtavus

Peida (*Restrict visibility*)

Muuda nähtavust, vajutades objekti real Ülevaate paremasse serva jäävat “silma” ikooni. See on kasulik keerukamate stseenide puhul, kui sa ei soovi objekti mõnele teisele kihile tõsta. See kehtib ainult objektidele, mis on nähtavatel kihtidel - nähtamatul kihil olev objekt jääb nähtamatuks olenemata sellest, mida Outliner näitab. V vajutamine muudab kõigi ülevaateaknas valitud objektide nähtavust.

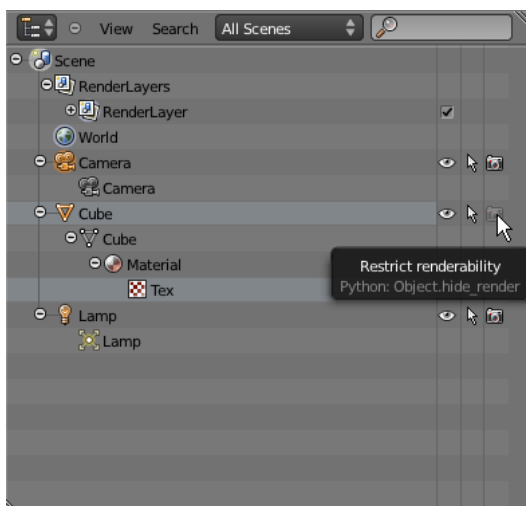
Valitavus



Keela valitavust

Lülita valitavust, vajutades “noole” ikoonile. See on kasulik, kui oled paigutanud mingi objekti stseeni ja ei soovi seda kogemata millegi muu kallal töötades valida. S vajutamine muudab kõigi Ülevaates eelvalitud objektide valitavust.

Renderduvus



Renderduvuse keelamine

Muuda renderduvust, vajutades “kaamera” ikoonile. See valik jätab küll objekti stseenis nähtavaks, kuid renderdusmootor hakkab seda ignoreerima. R vajutamine muudab kõigi Ülevaates valitud objektide renderduvust.

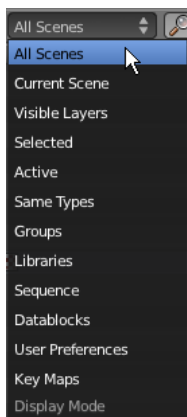
Otsimine

Failis olevaid andmeplokke saad otsida, kasutades otsingumenüüd (Search) Ülevaateakna päises või mõnda järgnevatest klahvikombinatsioonidest:

- F - Otsi lihtsalt Find.
- CtrlF - Otsi lihtsalt (tõstutundlik) Find (case sensitive).
- AltF - Otsi tervet fraasi Find complete.
- CtrlAltF - Otsi tervet fraasi (tõstutundlik) Find complete (case sensitive) .
- ⇧ ShiftF - Korda otsingut Find again.


Otsingusõnale vastavad andmeplokid (*datablock*) valitakse automaatselt.

Ülevaate nimekirja filtreerimine

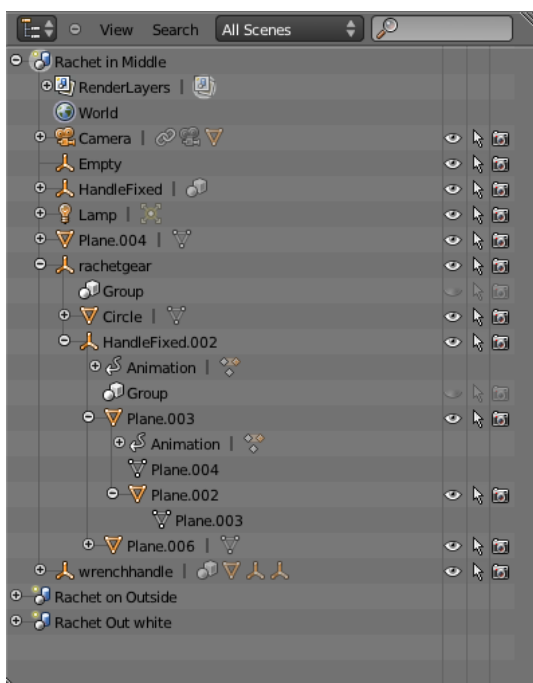


Ülevaate
filtreerimise
rippmenüü.

Akna päises on kast, millega saad valida, milliseid asju üldkontuur näitama peaks. Vaikimisi on kastis kõik stseenid. Võid valida, et näidataks ainult praegust stseeni, valitud andmeplokke, parjasti valitud kihtidel olevaid objekte jne. Need valikud on siin selleks, et saaksid *nimekirja lühendada* ning seeläbi asju kiiremini ja lihtsamalt leida.

- Kõik stseenid (All Scenes) - Näitab *kõike*, mida üldkontuur saab näidata (kõikides stseenides, kõikidel kihtidel jne)
- Praegune stseen (Current Scene) - Näitab kõike praeguses stseenis.
- Nähtavad kihid (Visible Layers) - Näitab kõike nähtavatel (parajasti valitud) kihtidel praeguses stseenis. Kasuta [kihinuppe](#), et muuta mõnel kihil asuvaid objekte 3D-aknas nähtavaks.
- Valitud (Selected) - Näitab ainult objekt(e), mis on parajasti 3D-aknas valitud. Mitut objekti saab korraga valida, vajutades nende peal ⇧ Shift RMB .
- Aktiivsed (Active) - Näitab ainult aktiivseid (sageli viimati valitud) objekte.
- Sama tüüpi (Same Types) - Näitab ainult praeguse stseeni neid objekte, mis on sama tüüpi kui parajasti 3D-aknas valitud objekt(id).
- Grupid (Groups) - Näitab ainult [gruppe](#) ja nende liikmeid.
- Teegid (Libraries) - VAJA TEHA
- Jadad (Sequence) - VAJA TEHA
- Andmeplokid (Data Blocks) - VAJA TEHA
- Kasutaja eelistused (User Preferences) - VAJA TEHA
- Nuppude kaardistused (Key Maps) - VAJA TEHA

Näide



Ülevaateaken näitamas nimekirja.

Näitepildil kujutatud .blend-failis on kolm stseeni: "Ratchet in Middle", "Ratchet on Outside" ja "Ratchet Out White". Vajutades nimest vasakul olevat väikest plussmärki, avab üldkontuur selle rea sisu ühe taseme võrra. Pildil on seda tehtud stseeni "Ratchet in Middle" puhul. Nagu näha võid, on stseenil mõned maailma ("World") materjali seaded, üks kaamera ("Camera"), üks tühi objekt ("Empty"), üks "HandleFixed"-nimeline objekt jne. Kõik need objektid on stseeni lisatud.

Vajutades "ratchetgear" kõrval olevat plussmärki, näeme, et see liigub, kasutades sissekande "Animation" kirjeldust; et see on loodud ringivõre "Circle" baasil ning et sellel on alamobjekt "HandleFixed.002", mis omakorda on peremeheks objektile "Plane.003" jne.

Kui nüüd valiksid mõne nendest andmeblokkidest, valitaks see ka 3D-aknas sellisel kujul, nagu see on võimalik. Vajutades * NumPad samal ajal, kui su hiire kursor on mõnes 3D-aknas, tsentreeritakse vaade sellele objektile. See tuleb kasuks. Samuti kustutab X vajutamine objekti ning töötavad ka kõik teised klaviatuuri kombinatsioonid, mida parajasti valitud tüüpi objektide puhul kasutada saab.

Lingitud teekide ülevaade

Blender suudab sirutada teistesse .blend-failidesse ja tõmmata nendest sisse kõike, mida tarvis. Nii toetab Blender mudelite taaskasutamist. Kui sul on näiteks teegina toimiv .blend-fail, milles on mingit tõsiselt lahe materjal, saad lahtiolevasse .blend-faili selle materjali lisada (Append). Nii ei pea sa käsitsi kõiki sätteid uuesti looma.

Üldine töömeetod

Mode: Kõik režiimid

Hotkey: ⇧ ShiftF1

Menu: File → Append or Link.






Blenderi peamenüü asub kasutaja eelistuste (User Preferences) aknas (vaikimisi sinu ekraani ülaseravas asuvas päises). Linkimiseni pääsemiseks pead valima menüüst File → Append or Link või vajutama aktiivses aknas ⇧ ShiftF1. Aktiivne aken muutub failisirvijaks (File Browser) (selle akna tüübi ikoon näeb välja nagu pruun kaust). Kasuta seda akent, et navigeerida oma kõvakettal või võrgus asuvatel ketastel olevate kataloogide ja alamkataloogideni, kus asub .blend-fail, mis sisaldab objekti, mida soovid taaskasutada. Vajutades .blend-failile (neile viitab nende nime kõrval olev oranž kast), läheb Blender selle sisse ja näitab sulle seal olevaid andmeblokkide tüüpe: stseene, objekte, materjale, tekstuure, võresid (vastavalt Scene, Object, Material, Texture, Mesh) jne. Ükskõik millisele neist vajutamine näitab failis olevaid seda tüüpi andmeblokke.

Kataloogide ja failide organiseerimine

Soovitame tekitada kataloogi /lib või /library. (Rahvusvahelise töö huvides on otstarbekas harjutada end kataloogipuud inglise keeles hoidma.) Selle teegi alla võid luua eraldi kataloogid igat tüüpi asjade jaoks, millele soovid ligipääsu ja võimalust taaskasutada (näiteks materials, textures ja meshes). Loo nendesse kataloogidesse omakorda alamkatalooge vastavalt sellele, kuidas su teek kasvab. Näiteks kataloogi võred võid soovida luua alamkataloogid inimesed, kosmoselaevad, mööbel, ehitised jne. Niiviisi tehes pead näiteks tooli võret sisaldava .blend-faili lihtsalt kopeerima kataloogi furniture.

Teegi objektide avatud projekti lisamine

Järgnev protseduur lisab objekti koos kõigi selle külge lingitud andmetega nagu võre, materjalid, tekstuurid jne avatud .blend-faili.

1. Vali menüüst File → Append or Link.
2. Otsi üles ja vali fail, mis sisaldab objekti, mida soovid lisada (sageli mõni teegifail).
3. Navigeeri selle faili Objektide osasse.
4. Vali üks objekt, kasutades LMB , mitu objekti RMB  abil ja/või järjestikused objektid RMB  all hoides ja hiirt lohistades.
5. Korda eelnevat igat tüüpi objektide puhul, mida soovid lisada. Ülemobjektid ja skeletid (kõik töötlejad) tuleb valida eraldi.
6. Vali päisest sobivad sätted (kursori kohale: At Cursor, aktiivsele kihile: Active Layer) vm.
7. Vajuta LMB  teegi laadimise nupul (Load Library) või vajuta ↵ Enter või MMB  otse lisatavatel andmetel.


Loomulikult saad lisada või linkida palju muud peale objektide: kogu objektide andmestiku (ObData) - kaamerad, kõverad, valgused, materjalid, võred jne - ja kas või **terve stseeni**. Pea meeles, et objekti ja objekti andmete (näiteks võre) lisamisel on **suur** vahe. Lisades võre andmebloki, tood sa sisse spetsiifilise üksikvõre ja mitte objekti, milles see võre on esindatud ja mida sa näha saad.

Kasuta akna File Browser päises olevat lisamise nuppu (Append), mis on vaikimisi valitud, kui soovid objektist enda faili sisse teha kohaliku iseseisva koopia. Vali linkimine (Link), kui soovid luua dünaamilise lingi alusfailiga - kui keegi muudab objekti algses failis, siis värskendatakse ka praegust faili, kui selle järgmine kord avad.

Vajuta teegi lisamise nupule (Load Library), et objekti oma praegusse .blend-faili lisada või seda sinna linkida.

Aknatüübi File Browser päises olevaist laadimisvõimaluste nuppudest olulisemad on:

Vali automaatselt (AutoSel)

Kui objekt laetakse, ei ole see aktiivne või valitud, vaid lihtsalt maandub sinu .blend-faili. Sageli soovid sa kohe pärast laadimist sellega midagi teha (näiteks muuta selle mõõtkaava või seda liigutada). Vali see nupp, kui soovid, et pärast importimist oleks objekt valitud, justkui oleksid ise maagiliselt sellele RMB  vajutanud. Selle nupu abil jääb ära vajadus objekt üles otsida ja see valida.

Aktiivsele kihile (Active Layer)

Blenderis on 20 kihti, mida kasutatakse suurte stseenide jagamiseks, ning iga objekt asub mõne(de)l neist. Vaikimisi laetakse objekt sinu faili sama(de)le kihile/kihtidele, nagu see oli algses failis. Objekti laadimiseks ainult praegusele aktiivsele kihile, millel töötad, vali see nupp.

Kursori asukohta (At Cursor)

Vaikimisi laetakse objekt sinu faili samale kohale, kus ta asub algses failis. Asetamaks seda laadimisele sinu poolt (3D-)kursoriga märgitud kohale, vali see nupp.



Laetu leidmine

Kui laetud objekt ei ole nähtav, soovime kasutada nuppu At Cursor või AutoSel. Kasutades nuppu AutoSel, pea meeles, et võid kursori selle objekti külge nakkuda (⇧ ShiftS4 (Cursor -> Selection)) ja tsentreerida sellele oma vaate (C (View → Align View → Center View to Cursor)). NB! Need tööriistad ei tööta, kui objekt asub kihil, mida sa pole valinud, sest mittevalitud kihtidel olevad objektid on nähtamatud.

Objektide taaskasutus (Võred, Kõverad, Kaamera, Valgused...)

Oletame, et sa lood ühte .blend-faili ratta ja soovid seda oma praeguses projektis uuesti kasutada. Ratta füüsiline mudel oleks võre ja koosneks eeldatavasti veljest ja rehvist. Loodetavasti andsid võrele mõistliku nime, võib-olla näiteks "Ratas". Sel rattal on võib-olla värv ja seetõttu on sellel määratud mõned materjalid (nagu kumm ja kroom).

Kui oled navigeerinud õige failini, vali "Ratas" (objekti andmeblokis) ja see imporditakse sinu praegusesse faili. Võid importida selle koopia või ka ainult sellele linkida.



Linkimine

Kui sa lingid faili ja seda hiljem muudad, siis näidatakse seda sulle praeguses failis sellisena nagu "see on" (muudetuna) alles siis, kui praeguse faili järgmine kord avad.

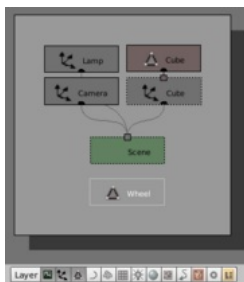
Kunstnikud on sageli muutnud oma mudelid vabalt kasutatavaks ning sõbrad võivad omavahel .blend-faile e-posti teel või kuhugi üles pannes jagada. Hoides neid faile, nagu ka oma varasemaid projekte, oma arvuti/serveri kataloogis [Download](#), ei pea sa enam kunagu uuesti ratas leiuatama.

Lingitud objektid on valituna ääristatud helesinise kontuuriga. Tavalised objektid on valituna ääristatud roosa kontuuriga.

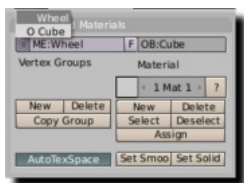
NB! Lingitud objekte ei saa liigutada! See paikneb samas kohas nagu algses failis. Et objekti liigutada või muuta selle mõõtkava/pööret, muuda see proksiobjektiks (*proxy object*).



Lisatud/lingitud võrede andmete kasutamine



Mõne ressursi (näiteks võre) lisamisel või linkimisel ei pruugi see kohe 3D-vaates nähtav olla. See tuleneb sellest, et andmed on küll Blenderisse laaditud, kuid neid pole mõnele objektile määratud ja need on seetõttu nähtamatud. Võid selle kindlaks teha, vaadates aknasse üldvaateaknasse Outliner ja muutes vaate tüübiks OOPS-skeem (OOPS Schematic, vajadusel pead akna päises valima stseeni andmebloki näitamise nupu Displays Scene datablock). OOPS-skeemi pildil näed, et "Ratas" ei ole lingitud ühegi objekti külge.



Lubamaks äsjalaaditud võre *Ratas* määramist mõnele objektile, vali kas mõni parajasti nähtav olev objekt või loo uus (näiteks üks kuup), mine seejärel materjalide linkimise paneeli (Link and Materials) ja vali võrede rippmenüüst võre *Ratas*, mille järel peaksid seda nägema, sest see on nüüd objektile määratud.

Kui sa võre Lisamise/Linkimise asemel laadid selle objekti Blenderisse, peaks see olema 3D-vaates kohe nähtav, ilma et peaksid seda mõne objektiga seostama (sest see on juba tehtud!).

Materjalide/tekstuuride sätete taaskasutamine








Materjali eelvaade pildibrauseris

Mõnesid materjale, nagu klaasi ja kroomi, on väga keeruline "täpselt õigeks" timmida. [Blenderi Sihtasutus](#) on näiteks väljastanud [Materjalide CD](#), mis on nende kodulehelt vabalt allalaaditav. Kasutades plaadil olevaid .blend-faile, saad importida igapäevaseid materjale nagu klaas, kroom, puit ja banaanid. See säästab palju aega, sest sageli ei pea sa tegelema nuppude ja liuguritega nokitsemisega, et mõnd materjali lihtsalt uuesti luua. Mainisin eraldi banaan materjali, kuna see on väga hea näide lihtsast protseduurmaterjalist, mis kasutab väga realistliku välimuse saavutamiseks värvüleminekut (ColorRamp) ja protseduurtekstuuri. Navigeerides õige failini ja valides Material, näitab brauser sulle materjali kerakujulist näidist, et saaksid tekstuuri nime ja välimuse enda jaoks kokku viia. Pildibrauseri (Image Browser) kohta lisainfo saamiseks vaata [väljalaskemärkmeid](#).

Blenderi lisa: Teek

On olemas ka fantastiline Pythoni skript nimega [Blenderi teek \(Blender Library\)](#), mis kogub kokku kõik sinu failid ja laseb sul koostada üldteegi. Skript näitab eelvaadet ning aitab sul oma tööd Blenderis organiseerida. Soovitame väga soojalt otsi lehel www.blendernation.com terminit "Blender Library", kuid see on leitav ka [Blenderi Wiki skriptide sektsioonist](#).

Sõlme asetuste taaskasutus

Sõlmeskeemi (ehk nuudlite) taaskasutamiseks ava originaal- ehk allikfail ja loo sõlmedest, mida arvad vajavat, üks grupp (Group). Kui soovid seda sõlmede gruppi oma praegusesse faili importida, vajuta lisamiseks LMB  valikul File → Append või linkimiseks LMB  valikul File → Link Info akna päises (või vajuta F1, et lisada (Append) või CtrlAltO, et linkida (Link)), ja navigeeeri vastava failini. Faili sisenedes näed sõlmepuu (NodeTree) valikut. Vajuta selle LMB  ja sulle näidatakse selles failis olevaid sõlmede gruppe. Vajuta LMB  neist sellel, mida soovid, ja seejärel vajuta LMB  teegi laadimise nupul .

[\[Verse\]](#)


Verse on imeline vabavaraline grupitöövahend, mis integreerub Blenderiga. Verse laseb mitmel inimesel reaajas Blenderi failidega töötada, neid linkida ning jagada nende objekte ja muudatusi.

Proksiobjektid

Proksi ('proxy') on inglise keeles juriidilises terminoloogias pärisobjekti asendaja või vahendaja; arvutiterminoloogias tähistab proksiserver veebilehtede kiiremaks kasutamiseks mõeldud puhverserverit. Kui sa teed Blenderis (ülalkirjeldatud) lingitud koopia, siis ei saa sa seda objekti muuta: sul on vaid link sellele. Sa ei saa sellele midagi lisada ega seda kohandada, sest see paikneb avamata failis.


Meeskonnana töötades võid aga soovida suuremat paindlikkust. Näiteks autot modelleerides võib üks inimene tegeleda auto kujuga (võrega) ja teine võimalike värvalikutega (materjalidega). Sellisel juhul soovid sa anda maalijale objektist proksi ja lubada tal selle materjalide sätteid muuta. Tavapärasemal juhul on sul tegelane, kelle animeerimisega tegeleb terve meeskond; nad saavad küll luua poose, kuid ei saa muuta tegelase värvi või skeletti, ehk siis nad saavad kasutada ainult seda, mida taageldaja on varem valmis ehitanud.

Proksiobjektide põhiomadus on see, et nad lubavad sul muuta andmeid kohalikult, kuid võimaldavad ka seda, et teatud andmete muutmine oleks keelatud. Andmed, mis on defineeritud keelatatuna, laaditakse teegist iga kord uuesti (tavaliselt faili lugemisel või ennistamisel/uuestitegemisel). Need keelud on paika pandud refereeritavas teegis endas, mis tähendab, et ainult teegis asuv fail saab defineerida seda, mida on kohalikult lubatud muuta.

Pooside puhul saad seda kontrollida, määrates luude kihi keelatuks. Keelatud kiht on tähistatud selles oleva musta täpiga. Vajuta nupul Ctrl LMB , et kihti keelata või taaslubada.

Mode: Objekti režiim

Hotkey: CtrlAltP

Et ise proksiobjekti luua, loo ülaloleva kirjelduse järgi kõigepealt link alusobjektile. Kui koopia on valitud (RMB ) ja vaates (st näed seda 3D-vaates), vajuta CtrlAltP ja kinnita valik proksi tegemise (Make Proxy) dialoogis. Objektile antakse originaali nimi ja “_proxy” sufiks. Nüüd saad vahendavat proksit liigutada ja muuta. Valituna näeb see välja nagu kohalik objekt (oranži kontuuriga).

Seejärel saad muuta keelustamata andmeid. Enamikel objektidel kuuluvad nende hulga asukoht ja pööre. Saad ka objekti asukohta lpo kõverate abil animeerida. Võrega objektidel on võre kuju keelustatud, nii et võtmevorme (*shape keys*) sa defineerida ei saa. Faili uuesti laadides värskendab Blender su faili koos kõigi originaali keelatud andmetes tehtud muudatustega, kuid ei tühista sinu muudatusi (kui just omanik seda ise pole teinud).

Skeletid ja mitmikesindajad

See funktsionaalsus on alles töös; Blenderi versioonis 2.43 ja CVSis (seisuga 29. aprill 2007), kontrollib proksiobjekt "kõiki grupi esindajaid (*instance*)". Pole võimalik luua ühte proksit igale grupi esindajale. See puudutab eriti seda, et ei ole võimalik luua üht proksi skeletti iga grupi esindaja kohta. Üks osaliselt efektiivne lahendus on objektide faili lisamine (*append*), mitte nendele linkimine, kui soovitakse esindajaid duplitseerida. Faili lisamise puhul originaali muutmisel lisatud objekte ei värskendata.

Kui sa kasutad POSIX-failisüsteemi, saad "üks proksi ühe grupi kohta" piirangust mööda hiilida, kasutades [artiklis "Linked Lib Animation Madness"](#) toodud juhiseid.

Sissejuhatus

Blenderit kasutades lood maailma neljas mõõtmes:

1. Vasak-parem, tavaliselt nimetatakse "x"-teljeks.
2. Ees-tagas, tavaliselt nimetatakse "y"-teljeks.
3. Üleval-all, tavaliselt nimetatakse "z"-teljeks.
4. Ajas muutuv, mis saavutatakse objektide, materjalide ning kaadritesse talletatud liikumise animeerimise abil.

Probleemiks on sinu ees olev kahemõõtmeline arvutiekraan! Oma hiirt saad liigutada ainult vasakule-paremale ja üles-alla. Sa ei saa minna ajas tagasi ning sul ei ole ka võimalik sirutada kätt läbi ekraani, haarata mõnd objekti ja seda kusagile mujale liigutada.

Selle asemel pead Blenderit juhendama nii, et ta seda sinu eest teeks. See peatükk sellest räägibki, kuidas Blenderi unikaalse kasutajaliidese toel oma virtuaalses maailmas ringi liikuda saad.

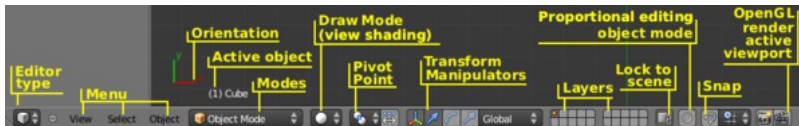
Sissejuhatus

3D-vaade (3D View) on koht, kus viid läbi enamuse objekti modelleerimisest ja stseeni loomisest. Blenderis on palju erinevaid tööriistu ja sätteid, mis aitavad sul hiire, klaviatuuri ja numbriklahvistiku abil edukalt tööd teha.

Samuti on 3D-vaade Blenderi vanim ning seega võimalusterohkeim ja enimseadistatav osa. Aga pole põhjust heituda. Niisiis, õpi omas tempos, paari sammu haaval, ja mängi pakutavate võimalustega, et näha, mida on võimalik teha.

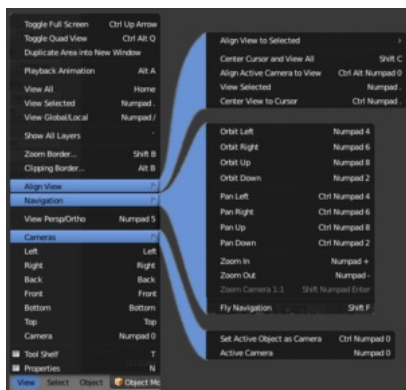
3D-akna päis

3D-vaate aken (3D View) koosneb tööpinnast ja päisest. Päis asub tööpinna all või selle kohal ning seda saab soovi korral peita. Päis näitab sulle menüüd ja kehtivat režiimi (seda seletatakse allpool).



3D-vaate päis.

Vaate menüü (View)



Vaate menüü (View).

Omaduste paneel (Properties Panel)

Näitab/peidab omaduste paneeli Properties (N), mille kaudu saad muuta erinevaid 3D-vaate sätteid:

- [Teisendamine \(Transform\)](#)
- [Rasvakriit \(Grease Pencil\)](#)
- [Vaade \(View\)](#)
- [Ese \(Item\)](#)
- [Esitus \(Display\)](#)
- [Taustapilt \(Background Images\)](#)
- [Teisendussuunad \(Transform Orientations\)](#)

Tööriistariiul (Tool Shelf)

Näitab/peidab tööriistariiulit Tool Shelf (T), mis tekib 3D-vaate vasakusse serva ning mille kaudu saad lähtuvalt valitud objekti tüübist ja kasutatavast režiimist läbi viia erinevaid tegevusi.

Kaamera (Camera, 0 NumPad)

Muudab vaate kaameravaateks.

Vaatenurgad

Need käsud muudavad vaate vaikimisi ülalt/alt, eest/tagant, paremalt/vasakul vaadeteks.

- Ülalt (Top, 7 NumPad)
- Alt (Bottom, Ctrl7 NumPad)
- Eest (Front, 1 NumPad)
- Tagant (Back, Ctrl1 NumPad)
- Paremalt (Right, 3 NumPad)
- Vasakult (Left, Ctrl3 NumPad)

Kaamerate menüü (Camera)

Muuda aktiivne objekt kaameraks (Set Active object as camera)
Aktiivne kaamera (Active camera)

Perspektiivne/paralleelne vaade (5 NumPad)

Need käsud lasevad sul muuta 3D-vaate projektsiooni

Navigatsiooni menüü (Navigation)

Siinses alammenüüs on käsud vaate pööramiseks ja liigutamiseks. Nende käskude kasutamine menüü kaudu ei ole eriti tõhus. Samas on siin, nagu ka kõigis teistes Blenderi menüüdes, toodud valikute kõrval ära need oluliselt mugavamad klaviatuuri kiirvalikud, mida otsid.

Vaate joondamine (Align View)

Selle alammenüü abil saad 3D-vaadet erinevatel viisidel joondada.

- Joonda valitule (Align to selected)
- Tsentreeri kursor ja vaata kõike (Center cursor and view all)
- Joonda aktiivne kaamera vaatega (Align active camera to view)
- Vaata valitut (View Selected)
- Tsentreeri vaade kursorile (Center View to cursor)

[Vaatekaugus](#) (*Clipping Border*, AltB)

Laseb sul sisestada vaatekauguse, mis piirab, millist osa kolmemõõtmelisest ruumist näed.

[Suurendamise ala](#) (*Zoom Border...*, ⇧ ShiftB)

Laseb sul valida piirkonna, mida soovid suurendada.

[Näita kõiki kihte](#) (*Showall Layers*, ~)

Muudab kõik kihid nähtavaks.

[Globaalne/kohalik vaade](#) (*Global View/Local View*, / NumPad)

Globaalne vaade näitab kõiki kolmemõõtmelises ruumis olevaid objekte. Kohtalik vaade näitab ainult valitud objekte. Sellest on abi siis, kui stseenis on palju objekte, mis jäävad teistele ette. Juhuslikult / NumPad vajutamist võib uuel Blenderi kasutajal suhteliselt sageli ette tulla, nii et kui saladuslikult terve hunnik objekte su stseenist kaduma läheb, proovi lihtsalt uuesti globaalsesse vaatesse tagasi minna.

[Vaata valitut](#) (*ViewSelected*, . NumPad)

Suurendab 3D-vaadet nii, et kõik *valitud* objektid selle sisse mahuks.

[Loe lähemalt 3D-vaate suurendamise kohta »](#)

[Vaata kõike](#) (*ViewAll*, ⇧ Home)

Suurendab 3D-vaate nii, et *kõik* praeguse stseeni objektid selle sisse mahuks.

[Animatsiooni taasesitamine](#) (*Play Back Animation*, AltA)

Paneb animatsiooni parajasti valitud kaadrist alates mängima.

[Duplitseeri piirkonda uues aknas](#) (*Duplicate area in newwindow*)

Duplitseerib praeguse 3D-vaate uues aknas

[Nelikvaade](#) (*Quad View*)

Näitab/peidab neljapaneelilist 3D-vaadet, millest igaüks näitab stseeni erineva nurga alt.

[Täisekraan](#) (*Toggle Full Screen*, Ctrl↑)

Täidab 3D-vaatega terve ekraani.

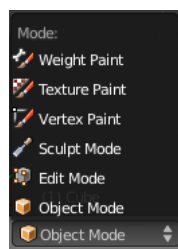
Valiku menüü (*Select*)

Selles menüüs on objektide valimise tööriistad. [Loe valimisest lähemalt »](#).

Objekti menüü (*Object*)

See menüü on nähtav, kui oled objekti režiimis. Muutmise režiimis olles muutub see tööriistu sisaldavaks menüüks. [Loe objektidest lähemalt »](#).

Režiimide nimekiri



Režiimide rippmenüü Mode.

Blenderis on mitu erinevat töörežiimi.

- Objekti režiimi (Object) abil saad töötada tervete objektidega.
- Muutmisrežiim (Edit) laseb sul muuta objekti kuju.
- [Voolimisrežiim](#) (Sculpt Mode)
 - Selles režiimis muutub su kursor tööriistaks, millega saad objekti voolida.

Kursor muutub pintsliks järgmistes režiimides:

- [Tippude maalimise](#) režiim (*Vertex Paint*)
- [Mõju maalimise](#) režiim (*Weight Paint*)
- [Tekstuuri maalimise](#) režiim (*Texture Paint*).

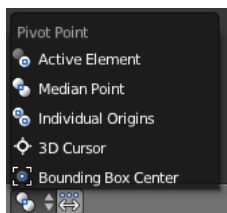
Vaateakna varjutuste nimekiri (*Viewport Shading*)

Selle abil saad muuta seda, kuidas sulle 3D-objekte vaateaknas kuvatakse. Vaata [järgmist lehekülge](#).

- Piirdkast (Bounding Box)
- Traatvõre (Wireframe)
- Tahke (Solid)
- Tekstuuriga (Textured)

[Loe 3D-vaate sätete kohta lähemalt »](#)

Keskpunkti valik



Keskpunkti valik.

Kui objekte või tippude/servade/külgede kogumeid pöörad või nende mõõtkava muudad, võid soovida nende keskohta (ehk teisendamise keskpunkti) kolmemõõtmelises ruumis liigutada. Kasutades seda menüüd, saad liigutada keskpunkti:

- Aktiivse objekti keskpunkti (Active Element)
- Mediaanpunkti (Median Point) - valitud objektide arvutuslikku keskpunkt
- Iga üksikobjekti keskpunkti (Individual Origins)
- 3D-kursori asukohta (3D Cursor)
- Piirkasti keskpunktid (Bounding Box Center)

Kasuta objekti keskpunkti nuppu Object Center, et valida, kas muudad tervet objekti või ainult objekti keskpunkti asukohta.

Keskpunktid on korralikult lahti kirjutatud [siin](#).

Teisendamise (manipulaatorite) valikud

Need lihtsad valikud lasevad sul objekte pöörata ja liigutada, haarates (ehk siis hiirega klõpsates) kinni nende kontrolleritest ning liigutades hiirt mööda mõnda telge. Need on korralikult lahti kirjutatud [manipulaatorite lehel](#).

Kihtide valimine

Kihid on ära seletatud [kihtide lehel](#). Täpsemalt on [kihtide vaatamise](#) osas lahti kirjutatud see, kuidas valida neid kihte, mida näha soovid, ning samuti räägitakse sellel lehel ka [objektide liigutamisest kihtide vahel](#).

Stseenile lukustamine

Kihtidest paremale jääv "luku" nupp on vaikimisi valitud. See tähendab, et valitud vaates on aktiivsed kihid ja kaamera samad, mis terves stseenis (ja seega kasutatakse neid renderdamisel). Seega kasutavad kõik 3D-vaated, mis on samamoodi lukustatud, samu aktiivseid kihte ja kaameraid — kui muudad neid ühes vaates, näidatakse seda muudatust kohe ka kõigis teistes vaadetes.

Kui selle "luku" nupu välja lülitad, saad iga vaate jaoks määrata erinevad aktiivsed kihid ja kaamera. See on kasulik näiteks siis, kui ei soovi, et su tööala (vaated) kogu stseeniga risustuks, aga soovid samas, et sul oleks kogu aeg silme ees ka üks tervikvaade (mis on lukustamata ning näitab näiteks kõiki kihte). Või selleks, et kasutada erinevates vaadetes erinevaid aktiivseid kaameraid (pea meeles, et kiirvaliku Ctrl0 NumPad vajutamine muudab aktiivse objekti aktiivseks kaameraks)...

Stseenidega töötamisest loe edasi [stseenide](#) lehelt.

Võre külge nakkumine (*Snap to Mesh*)

"Magnet" nupu abil saad kontrollida nakkumist, et objektide teisendamine ja modelleerimine lihtsamad oleksid. Loe sellest edasi [nakkumise](#) lehelt.

Renderdamise nupud

Renderdamise nupud renderdavad 3D-vaatest OpenGL-i versiooni.

Esimene nupp renderdab seisva pildi. See vastab põhimõtteliselt täpselt sellele, mida sa näed — ehk siis see eemaldab vaid ruudustiku/teljed/jms. See nupp kasutab sama joonistamise režiimi (Draw) kui 3D-vaade, nii et sellest nupust on kasu näiteks siis, kui keegi soovib näha töös oleva objekti traatvõret.

Teine nupp renderdab 3D-vaatest animatsiooni ja on seega kasulik animeeritud stseenide eelrenderdamiseks. Animatsioon salvestatakse renderdamise konteksti Render väljundi paneelis Output määratud kataloogi ja formaadis.

Sissejuhatus

Blender võimaldab sul kahemõõtmelisel ekraanil töötada kolmemõõtmelises ruumis. Selleks et kolmes mõõtmes töödada, peab sul olema võimalus muuta oma vaatenurka ning seda suunda, kust sa stseenile lähened. See on kõigis 3D-vaateakendes ka võimalik. Kuigi kirjeldame siin 3D-vaate akent (3D View), kasutab enamik teisi aknaid samasuguseid operatsioone. Näiteks saab suurendada ja nihutada asju ka nuppude aknas ja selle paneelides.



Hiireklahvid ja numbriklahvistik

Kui sul on vähem kui kolme klahviga hiir või klaviatuur ilma numbriklahvistikuta, uuri [klaviatuuri ja hiire](#) lehte ja uuri, kuidas Blenderit selliste seadmetega kasutada.

Perspektiivsed ja paralleelsed vaated

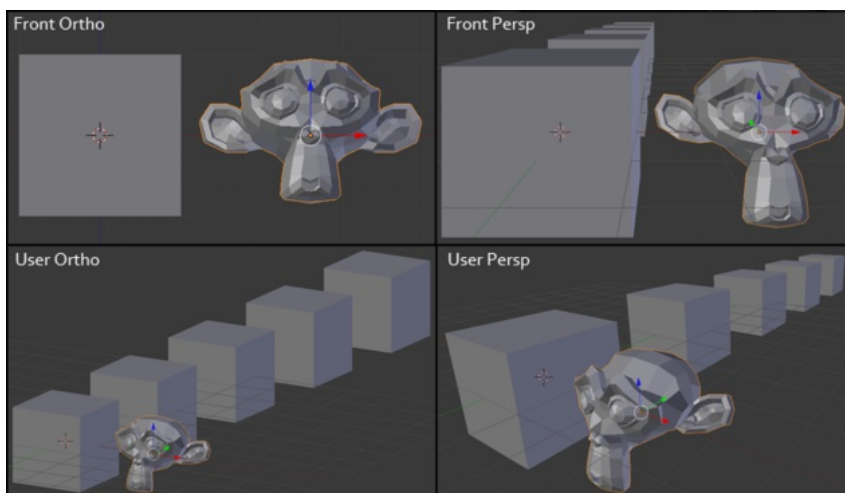
Mode: Kõik režiimid

Hotkey: 5 NumPad

Menu: View » Perspective / View » Orthographic

Kirjeldus

Iga 3D-vaateaken lubab kasutada kahte erisugust projektsioonitüüpi. Seda demonstreerib joonis (*Paralleelne (vasakul) ja perspektiivprojektsioon (paremal)*):



Paralleelne (vasakul) ja perspektiivprojektsioon (paremal).

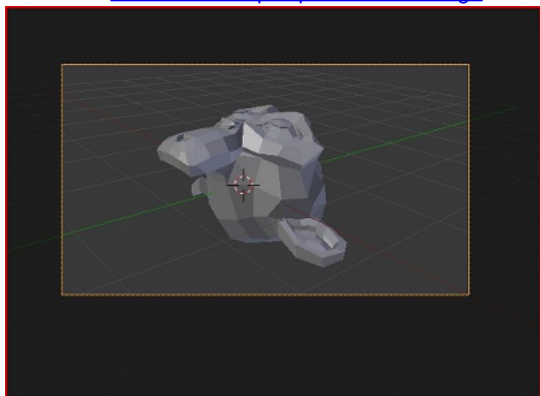
Meie silmad on perspektiiviga harjunud, sest kaugemad objektid näivad väiksematena. Paralleelprojektsioon (paralleelne vaade) näib esmapilgul imelik, sest objektide suurus ei olene nende kaugusest. Vaataksid stseeni justkui mõnest lõputult kaugest punktist. Igal juhul on paralleelne vaade väga kasulik (seda kasutatakse nii Blenderis kui ka enamikus teistes 3D-programmides vaikimisi), sest see näitab stseeni "tehnilisemalt" ning seeläbi on proportsioonide joonistamine ja nende täpsuse hindamine lihtsam.

Sätted

3D-vaate projektsioonitüübi muutmiseks vali kas menüüsissekanne paralleelne (View » Orthographic) või perspektiiv (View » Perspective). Kiirvalik 5 NumPad võimaldab liikuda nende kahe tüübi vahel.

Kaamera projektsioon

3D-vaate projektsioonitüübi muutmine ei mõjuta seda, kuidas stseeni renderdatakse. Vaikimisi renderdatakse perspektiiviga. Kui soovid renderdada paralleelprojektsioonis, vali kaamera ja vajuta selle paneelil Camera nuppu Orthographic muutmiskontekstis (F9) – vaata ka [seda renderdusperspektiivide lehekülge](#).



Kaameravaate näidis.

Valik View » Camera määrab 3D-vaate kaamerarežiimi (0 NumPad). Stseeni näidatakse nii, nagu seda hiljem renderdatakse (vaata

Kaameravaate näidist: renderdatav pilt sisaldaks kõike, mis jääb oranži punktiirjoone sisse. Selles vaates on võimalik suurendamine ja vähendamine, kuid vaatenurga muutmiseks pead sa kaamerat liigutama või pöörama.

Tehnilised detailid

Perspektiivi definitsioon

Perspektiiv ehitatakse geomeetriliselt üles järgnevalt: sul on kolmemõõtmeline stseen ja sina oled punktis o asuv vaataja. Stseeni kahemõõtmelise perspektiivi ehitamiseks asetatakse punkti o ette vaatamissuunaga ristuv tasapind (paberileht), millele see kahemõõtmeline perspektiiv siis joonistatakse. Kolmemõõtmelise stseeni iga punkti p puhul tõmmatakse joon po , mis läbib punkte o ja p . Lõikumiskoht s , kus po joon ja tasapind ristuvad, on perspektiivis projitseeritud punkt. Projitseerides kõik p punktid, tekib perspektiivis vaade.

Paralleelprojektsiooni definitsioon

Paralleelprojektsiooni puhul eksisteerib küll vaatesuund, kuid puudub vaatepunkt o . Joon tõmmatakse läbi punkti p , nii et see on vaatamissuunaga paralleelne. Joone ja tasapinna vaheline lõikumiskoht s ongi punkti p paralleelseks projektsiooniks. Projitseerides niimoodi kõik p punktid, saamegi kokku paralleelse vaate.

Vaate pööramine

Mode: Kõik režiimid

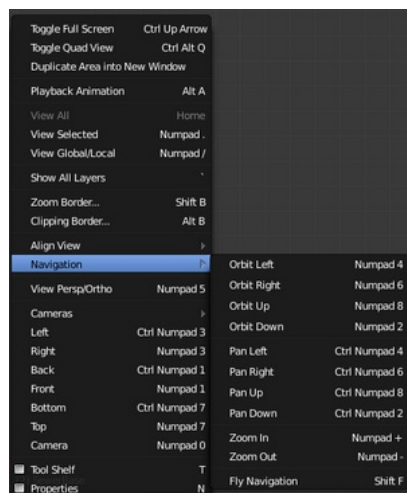
Hotkey: MMB  / 2 NumPad / 4 NumPad / 6 NumPad / 8 NumPad / CtrlAlt Wheel 

Menu: View → Navigation

Kirjeldus

Blender pakub vaatele vaikimisi nelja suunda: külgsaade (Side), eestvaade (Front), ülaltvaade (Top) ja kaameravaade (Camera). Blender kasutab täisnurkseid karteesia koordinaate, Z-teljega ülespoole. "Külgsaade" vastab X-telje kahanevas suunas vaatamisele, "eestvaade" Y-telje suunas vaatamisele ja "ülaltvaade" Z-telje suunas vaatamisele. Kaameravaade (Camera) näitab avatud stseeni kaamera vaatepunktist.

Seaded







3D-vaateakna vaatemenüü (View).

3D-vaateakna vaate suunda saad muuta vaatemenüü View valikutega (vt näidispilti *3D-vaateakna vaatemenüü (View)*) või vajutades kiirvalikuid - 3 NumPad "külgsaade", 1 NumPad "eestvaade" ja 7 NumPad "ülaltvaade" jaoks. Vastandvaateid saad valida samu numbriklahvistiku kiirvalikuid tehes nuppu Ctrl all hoides. Lisaks näeb kiirvalikuga 0 NumPad "kaamera" vaatepunkti.

Kiirvalikud

Pea meeles, et kiirvalikud mõjutavad **aktiivset akent**, mis on parajasti fookuses. Seega kontrolli, et su hiire kursor oleks alas, milles soovid töötada, enne kui teed kiirvaliku!

Lisaks vaikimisi suundadele, võib vaadet pöörata ükskõik millise nurga alla. Hoia all MMB  ja lohista hiirt vaateakna sees: kui alustad akna keskelt ning liigud üles-alla või vasakule-paremale, siis pöörleb vaade ümber akna keskpunkti. Teise võimalusena võid vajutada nuppu Alt samal ajal, kui hoiad hiireklahvi LMB  ja lohista hiirt vaateakna sees.

Vaatenurga muutmiseks tilukeste sammude haaval kasuta nuppe 8 NumPad ja 2 NumPad (mis vastavad ükskõik millises vaatepunktis MMB  all hoides üles-alla lohistamisele) või 4 NumPad ja 6 NumPad (või CtrlAlt Wheel ) , et pöörata ennast ümber globaalse Z-telje, *ükskõik milline vaatepunkt ka parajasti ei oleks*.

Juhtkuul/plaadimängija

Kui sa oma vaadet eelpoolsetes sektsioonides toodud juhiste järgi pöörad, siis vaikimisi pööratakse stseeni niimoodi, nagu liigutaksid oma kätt "**juhtkuulil**" (*trackball*). Mõnede kasutajate jaoks on see intuiitiivne ja mõnede jaoks mitte. Kui sulle tundub, et sedasorti 3D-akna pööramine on sinu jaoks keeruline, võid pöörata hoopis "**plaadimängija**" stiilis.

Plaadimängija stiil (Turntable) samaneb rohkem grammofoniga, mille puhul on kaks võimalikku pööramistele ning mille maailma “üleva” ja “all” on pealtnäha paremini defineeritud. Plaadimängija (Turntable) stiili kasutamise miinuseks on see, et kaotad objektidega töötamisel veidi paindlikkust. Samas saad juurde kindla “ülemise” ja “alumise”, mis võib sinu segadust mõnevõrra vähendada. Loomulikult võid parajasti käsileolevast tööst sõltuvalt nende stiilide vahel hüpata.



Vaate pööramisstiil.

Pööramise “stiili” muutmiseks kasuta [kasutaja eelistuste akent](#) (User Preferences). Vajuta sisendinupule Input, et minna lehele, millelt leiad erinevad *sisendite* teemalised seaded. Seal näed ala, mis tegeleb pöörlemisstiiliga (*Orbit*). Kasutaja eelistuste (User Preferences) vahekaardil Interface on veel kaks märkekasti, millest sõltub 3D-akna kuvamine. Automaatne perspektiiv (Auto Perspective) lülitab vaate automaatselt perspektiivprojektsiooni, kui seda MMB abil pööratakse. Valik Rotate Around Selection pöörab vaadet ümber parajasti valitud objekti keskpunkti. Kui midagi prajasti valitud ei ole (nt kui sa kasutasid kiirvalikut A, et kõik valitu tühistada), kasutatakse pööramiseks ikkagi viimati valitud objekti keskpunkti.

Vaate nihutamine

Mode: Kõik režiimid

Hotkey: ⇧ Shift MMB / Ctrl2 NumPad / Ctrl4 NumPad / Ctrl6 NumPad / Ctrl8 NumPad / ⇧ ShiftAlt LMB

Menu: View → Navigation

Kirjeldus

Vaate liigutamiseks ekraaniga paralleelselt vajuta nuppu ⇧ Shift ja lohista hiirt klahvi MMB all hoides 3D-vaateaknas. Tilkukeste sammudega liikumiseks kasuta kiirvalikuid Ctrl8 NumPad, Ctrl2 NumPad, Ctrl4 NumPad ja Ctrl6 NumPad nagu pööramise puhulgi (märkus: nupu Ctrl võib asendada nupuga ⇧ Shift). Keskmise hiirenuputa kasutajad võivad vajutada korraga ⇧ Shift ja Alt ning lohistada, hoides all LMB .

Vaate suurendamine

Mode: Kõik režiimid

Hotkey: Ctrl MMB / Wheel / + NumPad / - NumPad

Menu: View → Navigation

Kirjeldus

Suurendada ja vähendada saab, vajutades Ctrl ja lohistades hiirt, klahvi MMB all hoides. Kiirvalikud on + NumPad ja - NumPad. Need valikud on leitavad ka menüüs View → Navigation (vt näidisilti *3D-vaateakna vaatemenüü (View)*).

Kui sul on kerimisrattaga hiir, saad + NumPad ja - NumPad vajutamise asemel suurendada ja vähendada ratast kerides (Wheel). Kuna nuppude aknas (Buttons) on niivõrd palju paneele, liigutab ratta kerimine akent horisontaalselt vasakule ja paremale. Sedasi saad liikuda kitsastel või väiksematel ekraanidel (või siis, kui aken on kitsas) vajaliku paneelini. Alternatiivina võid muuta nuppude akna verikaalseks ning selle paneelid reastatakse ülalt alla.

Kui eksid teelt...

Kui sa 3D-ruumis enda asukoha kaotad, mis polegi niivõrd ebatavaline, siis aitavad sind kaks kiirvalikut: Home muudab vaate selliseks, et näed kõiki objekte (sama teeb View → View All menüüvalik) ja NumPad liigub ainult parajasti valitud objektide peale (sama teeb View → View Selected menüüvalik).

Suurendusala

Suurendusala tööriist (Zoom Border) lubab sul määrata nelinurkse ala ja seda suurendada niimoodi, et see täidaks terve 3D-vaate.

Tööriista kasutamiseks leia see menüüst View või kasuta kiirvalikut ⇧ ShiftB ja tõmba seejärel ekraanile ristkülik, mida soovid suurendada.

Vaate joondamine

Joonda vaade

Need sätted lasevad sul vaadet eri viisidel joondada ja suunistada. Valikud asuvad vaatemenüüs (View Menu)

Valitud objektile joondamise menüü (Align View to Selected)

Need valikud joondavad su vaate vastavalt valitud objekti koht-teljele (*local axis*) või muutmisrežiimis (Edit) valitud pinnanormaalile.

Ülalt (Top) ⇧ Shift7 NumPad
 Alt (Bottom) ⇧ ShiftCtrl7 NumPad
 Eest (Front) ⇧ Shift1 NumPad
 Tagant (Back) ⇧ ShiftCtrl1 NumPad
 Parevalt (Right) ⇧ Shift3 NumPad
 Vasakult (Left) ⇧ ShiftCtrl3 NumPad

Kursori tsentreerimine ja kõige vaatamine (Center Cursor and View All), ⇧ ShiftC

Liigutab kursori algpunkti ja suurendab/vähendab vaadet niimoodi, et näeksid tervet oma stseeni sisu.

Kaamera joondamine (Align Active Camera to View), CtrlAlt0 NumPad

Asetab aktiivse kaamera sinu hetke vaatepunkti

Valikule joondamine (View selected) . NumPad

Fokuseerib vaate parajasti valitud objekti(de)le, tsentreerib need vaateaknas ja suurendab neid, kuni nad täidavad terve ekraani.

Kursorile tsentreerimine Center view to cursor, Ctrl. NumPad

Fokuseerib vaate 3d-kursorile.

Valikule joondamine (View Selected)

vt ülalpool

Kõige näitamine (View All), ^ Home

Kadreerib kõik stseeni objektid nii, et nad oleksid vaateaknas nähtavad.

Kohalik ja globaalne vaade

Kohaliku (Local) ja globaalse (Global) vaate vahel saad valida vaatemenu View Menu kaudu või kasutades kiirvalikut / NumPad. Kohalik vaade eraldab valitud objekti või objektid nii, et nad on ainsad, mida vaateaknas näeb. See on kasulik, kui töötad objektidega, mida teised objektid varjavad või millel on keeruline geomeetria. Vajuta nuppu / NumPad, et globaalvaatesse tagasi saada

Vaatekauguse ala

Mode: Kõik režiimid

Hotkey: AltB

Menu: View » Set Clipping Border

Kirjeldus

Aitamaks sind keeruliste mudelite ja stseenidega töötamisel, võid muuta vaatekaugust, et visuaalselt eraldada see, mille kallal töötad.

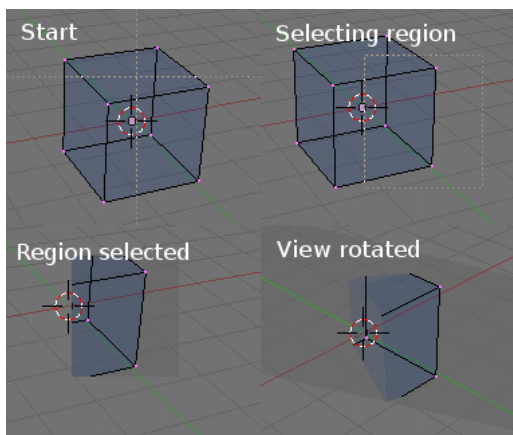
See tööriist näitab ainult seda, mis jääb sinu poolt defineeritud ala sisse.

Olles tööriista eelneval kiirvalikuga AltB aktiveerinud, pead soovitud 3D-vaates hiirega risküliku joonistama. Loodavaks vaatekauguse alaks on seejärel:

- Paralleelprojeksioonis vaate puhul (piiramatu kõrgusega) täisnurkne rööptahukas.
- Perspektiivse vaate puhul (piiramatu kõrgusega) ruudukujulise põhjaga püramiid.

Vaatekauguse ala kustutamiseks vajuta uuesti AltB.

Näide



Piirangu regioon ja maht.

(*Piirangu regioon ja maht*) näitab vaatekauguse tööriista kasutamist kuubi puhul. Aktiveeri tööriist kiirvalikuga AltB (vt pildi vasakul ülemises nurgas olevat sildiga "Start" märgistatud osa). välja Selle peale tekib punktiirjoonega sihiku kursor. Hoia all LMB ja lohista pildi paremas ülemises nurgas näidatud riskülikukujuline ala. Seeläbi määrad sa regiooni ning 3D-ruumis luuakse vastavalt sellele vaatekauguse maht. Pane tähele, et osa kuubist on nüüd nähtamatu ehk siis vaatekaugusest väljas. Kasuta MMB vaate pööramiseks ning näed, et nähtav on ainult see, mis jääb püramiidikujulise mahu sisse. Kõik muutmise tööriistad töötavad endiselt tavapäraselt, kuid ainult selle püramidaalse vaatekauguse mahu siseselt.

Pildil näidatav hall ala ongi vaatekauguse maht. Kui vaatekaugus uuesti kiirvalikuga AltB välja lülitada, muutub terve 3D-ruum uuesti nähtavaks.

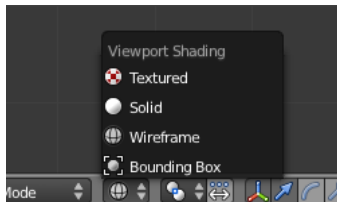
Vaate varjutmine

Mode: Kõik režiimid

Hotkey: Z / ⇧ ShiftZ / AltZ / ⇧ ShiftAltZ / D

Kirjeldus

Sõltuvalt arvuti kiirusest, stseeni keerulisusest ja parajasti tehtava töö olemusest võid valida endale sobiva joonistusrežiimi:



3D-vaate joonistusrežiimi nupp.

Textured (tekstuuriga)

Näitab UV-pilttekstuuriga mudeleid OpenGL-i valgustusega. Protseduur-tekstuure ega mitte UV-na laotatud tekstuure ei näidata.

Shaded (varjutatud)

Arvutab iga tipu jaoks ligilähedase tekstuuri ja valgustuse ning loob nende vahele sujuvad üleminekud. Meetod on oluliselt vähem täpne kui tekstuuride renderdusmootoriga kontrollimine, kuid samas on see kõvasti kiirem. Juhul kui stseenis puudub valgus, on kõik objektid mustad.

Solid (tahke)

See on vaikimisi joonistusrežiim, milles pinnad esitatakse vaid värvitoonidega, kasutades sisseehitatud OpenGL-i valgustust. See režiim ei sõltu stseeni valgusallikatest ning seda saab muuta kasutajaeelistuste akna (User Preferences) vahekaardil System & OpenGL nimetuse Solid OpenGL lights alla koondatud valikutega.

Loe lähemalt [sellelt lehelt](#).

Wireframe (traatvõre)

Objektid koosnevad vaid nende piirjooni esitavatest joontest (nt võrede või pindade servad...).

Bounding Box (piirdkast)

Objekte ei joonistata üldse. Selle asemel näitab režiim ainult risttahukaid, mis vastavad objektide suurusele ja kujule.

Nende režiimide vahel saad valida:

- Kasutades andmetüübi (Draw type) rippnimekirja 3D-vaate päises (vaata pilti *3D-vaate joonistusrežiimi nupp*).
- Vajutus D esitab joonistusrežiimi menüü (Draw mode).
- Kasutades Z-põhinevaid kiirvalikuid, nagu allpool kirjeldatud:

Joonistusrežiimid ja Z-põhised kiirvalikud.

Z	Vahetab joonistusrežiimi traatvõre (Wireframe) ja tahke (Solid) vahel.
⇧ ShiftZ	Vahetab joonistusrežiimi traatvõre (Wireframe) ja varjutatu (Shaded) vahel.
AltZ	Vahetab joonistusrežiimi tahke (Solid) ja tekstuuriga (Textured) vahel.
⇧ ShiftAltZ	Muudab vaate tekstuuriga (Textured) joonistusrežiimiks.

Vaate omaduste paneel

Mode: Kõik režiimid

Panel: View Properties

Menu: View » View Properties...

Kirjeldus

Lisaks ülalkirjeldatud päise seadistustele laseb vaateomaduste (View Properties) paneel muuta ka teisi 3D-vaadet puudutavaid sätteid. Selle avamiseks navigeeri menüüs View » View Properties....

Vaade (View)

Lens (läätis)

Määra 3D-vaate kaamera fookuskaugus millimeetrites (erineb [renderduskaamera](#) vastavast seadistusest).

Lock to Object (lukusta objektile)

Sisestades objektilahtrisse Object mõne objekti nime, lukustad vaate sellele – st objekt on alati vaate keskpunktis (ainus erand siin on aktiivne kaamera 0 NumPad). Kui lukustatud objekt on skelett, saad täpsustada lukustamist mõnele konkreetsele luule, sisestades lahtrisse Bone selle nime.

Lock to Cursor (lukusta kursorile)

Lukusta vaate keskpunkt 3D-kursori asukohale.

Lock Camera to View (lukusta kaamera vaatele)

Olles kaameravaates, saad kasutada seda valikut 3D-kaamera ruumis liigutamiseks. Kui vaadet liigutad, liigub kaamera sellega kaasa.

Clip Start ja Clip End (vaatekaugus)

Määrab lähima ja kaugema punkti, mida 3D-kaamera näitab.

Local Camera (kohalik kaamera)

Selles vaates kasutatav aktiivne kaamera.

3D Cursor Location (3D-kursori asukoht)

Siin saad määrata täpse 3D-kursori asukohta.

Ese (*Item*)

See jaotus näitab parajasti valitud objekti.

Esitus (*Display*)

Only Render (ainult renderdatavad)

Näitab ainult neid objekte, mida renderdatakse.

Outline Selected (valiku kontuur)

Kui see valik on tühistatud, ei näidata tahkes/varjutatud/tekstuuriga joonistusrežimis valitud objektide ümber enam värvilist kontuuri.

All Object Origins (kõik keskpunktid)

Sisselülitatuna näidatakse alati objektide lähtepunkti ja seda ka objektide puhul, mis pole valitud (vaikimisi võivad tahkes/varjutatud/tekstuuriga joonistusrežimis objektide lähtepunktid ülejäänud kujundite taha peituda).

Relationship Lines (seosjooned)

Otsustab, kas alluvussuhete, piirajate, kinnitamise jms punktiirjooni näidatakse või mitte.

All Edges (kõik servad)

Kui objekti kontekstis Object on servade pealiskihit sisse lülitatud, muudab see valik kõik traatvõred (*wireframe*) vaateaknas nähtavaks.

Grid Floor (põrandaruudustik)

Väljalülitatuna ei näidata sulle teistes vaadetes peale paralleelse ülalt/eest/külgvaate ruudustikku.

X Axis, Y Axis, Z Axis (teljed)

Otsustab, milliseid telgesid näitakse sulle teistes vaadetes peale paralleelse ülalt/eest/külgvaate.

Lines (joonte arv)

Otsustab mõlemat pidi ruudustiku moodustavate joonte arvu teistes vaadetes peale paralleelse ülalt/eest/külgvaate.

Scale (mõõtkava)

Määrab põrandaruudustiku mõõtkava.

Subdivisions (zlamjaotused)

Määrab, mitu alamjoont ilmub suurendamisel igasse ruudustiku kasti (seega kehtib see säte vaid paralleelprojektsioonis ülalt/eest/külgvaatele).

Shading (varjutus)

Määrab, kuidas objekte 3D-vaates varjutatakse.

Textured Solid (tekstuuriga tahked)

Näitab tahkes vaates külgedele määratud tekstuure.

Toggle Quad View (nelikvaade)

Lülitab neljaaknalist 3D-vaadet sisse ja välja. Vaata [akende paigutust](#)

Taustapilt

Mode: Kõik režimid

Panel: Background Image

Menu: View » Properties...

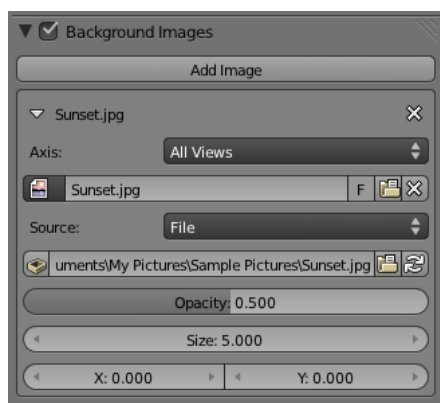
3D-vaatesse taustapildi lisamine on mitmel puhul äärmiselt kasulik: mõistagi modelleerimisel, kuid ka maalimisel (võid näiteks otse mudelile maalimisel nägude näidiseid kasutada...), animeerimisel (kasutades taustaks videot) jne.



Paari asja tuleb taustapiltide puhul silmas pidada:

- Need on nähtavad ainult omas aknas (st sul võib olla iga 3D-vaate jaoks erinev taust, näiteks ülalt/eest/kõrvaltvaade vastava akna jaoks).
- *Neid saab kasutada ainult paralleelprojektsiooni puhul ülalt-, kõrvalt- ja eestvaates (ning nende vastandvaadetes)! Pilt jääb nende kuue vaate vahel liikumisel samaks.*
- Tausta suurus sõltub akna suurendusest (st suurendamisel muutuvad nad suuremaks jne).
- Võid kasutada videofaile ja animeeritud pildiridasid.

Seaded



Taustapiltide paneel.

Taustapildi haldamine käib Blenderis vaatesätete paneelis (N) asuva taustapildi (Background Image) menüü abil. Selle paneeli ülaosas olev nupp lülitab taustapilti sisse ja välja. Vaikimisi on olemas koht ainult ühele pildile. Sätetesse saab siseneda vajutades hiireklahviga LMB valgele kolmnurgale.

Pärast sisselülitamist saab lisada pilte olemasolevast andmeblokist või uut faili laadides. Teljemenüü (Axis) määrab, millistes vaadetes on pilt nähtav. Piltide arvu saab muuta vajutades klahviga LMB nupule Add Image. Pärast pildi laadmist saab muuta järgnevaid sätteid:

Source (allikmaterjal)

Määrab failitüübi. Vastavalt valitud tüübile ilmub allapoole lisavalikuid:
File (fail)

Kasutab pildifaili
Source File (algfail)

Näitab reaalselt faili, millele antud andmebloki puhul lingitakse

Sequence (pildijada)

Numereeritud pildifailide jada
Frames (kaadrid)

Määrab, mitut pildifaili jada puhul kasutada

Start (algus)

Määrab esimese kaadri numbri

Offset (nihe)

Nihutab seerias kasutatavat kaadri numbrit

Fields (hulk kaadris)

Määrab, mitut pilti/välja ühe renderdatava kaadri puhul näidatakse (nt telepilt kasutab ülerealaotust)

Auto Refresh (zutomaatne uuendamine)

Värskendab kaadri vahetamisel automaatselt ka tausta

Cyclic (tsükiline)

Jätkab pärast lõppu jõudmist uuesti algusest

Movie (video)

Kasutab videofaili:
Match Movie Length (pikkuse ühtsustamine)

Määrab, mitmele kaadrile video paigutada

Generated (genereeritud)

Kasuta Blenderi poolt genereeritud pilti:
Width, Height (suurus)

Määra pildi kõrgus ja laius pikslites

Blank (tühi)

Genereerib tühja faili

UV Grid (UV-ruudustik)

Genereerib UV-laotuste testimise ruudustiku

Color Grid (värviruudustik)

Genereerib UV-laotuste testimiseks värvilise ruudustiku

Opacity (läbipaistvus)

See nihik määrab taustapildi läbipaistvust (alates täielikult nähtavast sättest **0.0** kuni täiesti nähtamatu sätteni **1.0**).

Size (suurus)

Määrab pildi suuruse (möötkava) 3D-vaates (Blenderi ühikutes).

X Offset, Y Offset (nihe)

Pildi horisontaalne ja vertikaalne vaateaknas liigutamine Blenderi ühikutes (vaikimisi on pilt keskel).



Kasuta madalaresolutsioonilist asendusfaili

Taustapiltide kasutamisel võid olla sunnitud arvuti võimekuse parandamiseks kasutama madalaresolutsioonilisi asendusfaile. Kui su ekraani resolutsioon on 800×600 , siis ei pea täisekraanil suurendamata olekus taustapilt olema sellest suurem. Kui su näidispilt on 2048×2048 , siis tegeleb su arvuti palehigis pikslite minemaloopimisega. Võta pilt ja vähenda seda (kasutades Blenderit või Gimpit) näiteks suurusele 512×512 . Jõudlus paraneb nii kuusteist korda ja samas ei kaota sa märgatavalt pildikvaliteedis. Mudelit detailsemaks tehes võid ka pildi resolutsiooni suurendada.

Kiirvalikud

Kaameravaade

Kaameravaade on üks stseeni vaatamise viise. Kaameraid kasutatakse võtete virtuaalseks komponeerimiseks ja selleks, et näha, milline on stseen renderdamisel.

Et muuta vaateakna vaade kaamera vaatenurgaks, vajuta 0 NumPad. Kaamera vaates nähtav on sama, mis lõplikul renderdusel nähtav. Kaameraid saab liigutada nagu kõiki teisi 3D-objekte.

Rohkem informatsiooni kaamerate kasutamise kohta leiad [renderdamise](#) peatüki [kaamerate](#) osast.

Kihid

Mode: Objektirežiim Object

Panel: Suhete paneel Relations (objekti kontekstis Object)

Hotkey: M

Menu: Object » Move to Layer...

3D-stseenid muutuvad keerukuse kasvades sageli järjest segasemaks. Mõnikord võib kunstnik ka soovida täpset kontrolli selle üle, kuidas üksikuid objekte valgustatakse, ega taha, et ühe objekti valgus mõjutaks teisi lähedalasuvaid objekte. Sellel ja allpool mainitud muudel põhjustel on objektid paigutatud ühte või enamasse "kihti". Objektkihte kasutades saad sa:

- Muuta oma 3D-vaates ainult mõnede kihtide objekte nähtavaks, valides vastavad kihid 3D-vaate (3D View) päiseribas, mille tulemusel muutub ekraani värskendamine/ülejoonistamine kiiremaks, väheneb virtuaalse maailma visuaalne müra ning kiireneb tööprotsess.
- Juhtida, [millised valgustid ühte objekti valgustavad](#), sundides valgustit valgustama ainult oma enda kihil/kihtidel asuvaid objekte.
- Kontrollida, millised jõud milliseid [osakeste süsteeme](#) mõjutavad, sest osakesi mõjutavad ainult nendega samal kihil asetsevad jõud ja efektid.
- Määrata, milliseid kihte (ja seega milliseid objekte) renderdatakse ning millised omadused/kanalid on komposiitimiseks saadaval, kasutades selleks [renderduskihte](#).

Skeletid võivad muutuda väga keeruliseks, sest kasutavad erinevaid luude tüüpe, kontrollereid, lahendajaid, erivorme jne. Kuna see kõik toimub suhteliselt ahtas ruumis, võib skelett muutuda segaseks. Seetõttu on Blenderis ka ekstra skelettide (*armatures*) jaoks mõeldud kihid. Skeleti kihid on objektid kihtidega väga sarnased, sest ka nende puhul saad sa skeleti (taglase - *rig*) erinevate kihtide peale laiali jagada ning näidata vaid neid kihte, millel töötada soovid.

[Loe skeletikihtide kohta lähemalt »](#)



Kihtidega töötamine

3D-kihid erinevad nendest, mida sa võib-olla 2D-graafika programmidest tunnud: nad ei mõjuta joonistamise järjekorda ning on peamiselt olemas (välja arvatud ülal toodud erilised omadused) selleks, et saaksid oma stseene korrastada ja struktureerida.

Renderdamisel renderdab Blender ainult valitud kihid. Kui kõik sinu valgustid on kihil, mis *ei ole valitud*, ei näe sa oma renderdusel midagi peale objektide, mis on üldvalguse (*ambient lighting*) poolt valgustatud.

[Grupeerimine ja allutamine](#) on samuti moodused, millega saab üksteisega seotud objekte loogiliselt kokku koguda. Infot nende kohta saab vastavatest teemadest.

Kihtide vaatamine

Blenderis on kaskümmend kihti ning sa saad päises asuvate sildistamata väikeste nuppudega valida, milliseid neist näidatakse (vt pilti *3D-vaateakna kihinupud*). Näitamaks ainult ühte kihti, vajuta vastavale nupule klahviga LMB ; rohkem kui ühe valimiseks kasuta kombinatsiooni ⇧ Shift LMB  — selle kombinatsiooni kasutamine juba aktiivse kihi nupul tühistab kihi valiku.



3D-vaateakna kihinupud.

Kihtide 1 kuni 10 valimiseks klaviatuuriga vajuta 1 kuni 0 (klaviatuuri tavalisel numbrite real, ülemine rida nuppe), kihtide 11 kuni 20 valimiseks vajuta Alt1 kuni Alt0 (alumine rida). Nupu ⇧ Shift vajutamine korraga mitme valiku tegemiseks või valiku tühistamiseks töötab ka klaviatuuriga valimisel puhul.

Stseenile lukustamine

Vaikimisi on kihtide nuppudest paremale jääv luku nupp sisse lülitatud. See tähendab, et vaadatavatele kihtidele tehtavad muudatused on nähtavad ka kõigis 3D-vaadetes, mis on samamoodi stseenile lukustatud – vaata ka [3D-vaates navigeerimise lehekülge](#).

Mitu kihti

Objekt võib olla korraga mitmel kihil. Näiteks lamp, mis valgustab ainult neid objekte, millega ta kihte jagab, võib "olla" kihtidel 1, 2 ja 3. Objekt kihtidel 3 ja 4 oleks valgustatud, kuid objekt kihtidel 4 ja 5 ei oleks. On palju kohti, kus tuleb tegemist teha kihispetsiifiliste efektidega — eriti valguste ja osakeste puhul.

Objektide kihtide vahel liigutamine



Kihi valik.

Valitud objektide liigutamiseks mõnele teisele kihile vajuta nuppu M ja vali siis ilmuvast dialoogist soovitud kiht. Objektid võivad samaaegselt olla rohkem kui ühel kihil. Objekti liigutamiseks mitmele kihile hoia hiirega klõpsamise ajal all nuppu ⇧ Shift.




Objekti konteksti Object valimine.

Teine moodus valitud objekti kihi muutmiseks on suhete paneeli Relations kaudu, mis asub objekti kontekstis Object.



Kihid objekti konteksti Object suhete paneelil Relations.

Seal asuval suhete paneelil Relations on kihtide nupud — nagu ennegi, saab ka nüüd objekti näidata enam kui ühel kihil, kui kasutada valimiseks kombinatsiooni ⇧ Shift LMB .

Kihtide animeerimine

Objekti kihilist “kuuluvust” [saab animeerida](#). Näiteks saavad objektid stseeni äkitselt tekkida ja sealt kaduda.

Objektide kihtidele jagamise näide

Soovituslik on kasutada ülemist nuppude rida oluliste asjade ning alumist vähekasutatavate või -muudetavate asjade või ülemise rea alternatiivide jaoks. Peamiselt kahele tegelasele keskenduv stseen võiks oma kihid jagada näiteks nii:

1. Peategelane.
2. Kõrvaltegelased.
3. Statistid (taustal liikuva tegelased).
4. Osakesed ja efektid (keerised, tuul).
5. Peamine tegevuskoht.
6. Peamised taustad ja dekoratsioonid.
7. Peamised rekvisiidid (lauad, toolid).
8. Väikesed rekvisiidid, maketid, dekoratsioonid, kaunistused.
9. Kaamerad, valgused.
10. Peategelase skeletid.
11. Kõrvaltegelaeste skeletid.
12. Statistide skeletid.
13. Vahetusriided.
14. Töös olevad võred.
15. Teised lavapaigutused ja mõõtmed.
16. Teised taustad, mida võib-olla kasutame.
17. Muud suured rekvisiidid, mis võiksid stseeni segaseks ajada.
18. Töös olevad rekvisiidid.
19. Lisavalgus.

Lokaalne või globaalne vaade

Mode: Kõik režiimid

Hotkey: / NumPad

Menu: View » Local View või View » Global View

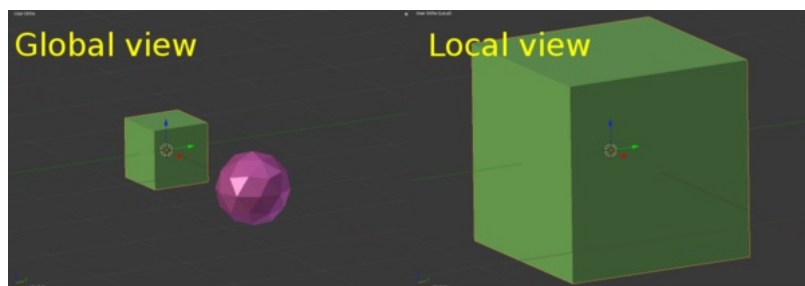
Kirjeldus

Lokaalses vaates (Local) olles näidatakse ainult valitud objekte, mis võib keeruliste stseenide muutmist lihtsustada. Lokaalsesse vaatesse sisenemiseks vali esmalt soovitud objektid ja kasuta seejärel menüükirjet View » Local View. Menüükirje View » Global View viib sind globaalsesse vaatesse tagasi. Nupp / NumPad vahetab vaadet nende kahe võimaluse vahel.

NB! 3D-vaate (3D View) päises olevad kihi ja lukustamise nupud on lokaalses vaates kadunud.

Näited

Globaalses vaates on kõik objektid nähtavad ja roheline kuup valitud. Lokaalsesse vaatesse minnes (/ NumPad) on kuup 3D vaates fokuseeritud ja vaate keskel. Kui mõnes stseenis on tuhandeid nähtavaid objekte, muudab see funktsioon programmi kasutamise kiiremaks, kuna vaid paar objekti jääb nähtavale.



Globaalne ja lokaalne vaade

Teisendused

Oma virtuaalses maailmas saad valitud objektiga teha mitmeid erinevaid asju. Võid seda väänata, muuta suuremaks või väiksemaks, seda ringi pöörata, selle kuju muuta ja nii edasi. See peatükk seletab sulle, kuidas kõike seda oma objektidega teha.

Kasutatavate teisenduste nimekiri on järgmine:

- [Advanced](#)
 - [Mirror](#)
- [Basics](#)
 - [Grab](#)
 - [Rotate](#)
 - [Scale](#)

Saad teisendusi sooritada kasutades:

- **kiirklahve**

Vajuta tegevuse kinnitamiseks klahvu

- **kontekstipõhiseid menüüsid**

Kui objekt on 3D-vaates objektirežiimis Object valitud, on menüüribal vaate (View), valimise (Select) ja objekti (Object) valikud. Vajuta objekti manipuleerimiseks valikut Object. Lahti hüppavas menüüs on teisendamise sissekanne Transform. Hiirekurssorit selle kohal hoides ilmub alammenüü, milles on loetletud võimalikud valikud objekti manipuleerimiseks (ja paremal pool ka vastavad kiirklahvid).

Lihtsad teisendused

Selles peatükis räägitakse 3D-vaate lihtsatest teisendustest ja manipulatsioonidest:

- [Elementide haaramine \(liigutamine\).](#)
- [Elementide pööramine.](#)
- [Elementide mõõtkava teisendamine.](#)
- [Elementide nakkumine teisendamise ajal millegi külge \(geomeetria, ruudustik, jne\).](#)
- [Teisenduste täpsuse kontrollimine.](#)
- [Teisenduste tegemine hiire liigutamise asemel numbrite sisestamise abil.](#)
- [Objekti teisenduste algseadistamine.](#)

Haaramine/liigutamine (*Grab/Move*)

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: G

Menu: Object/Mesh/jne. → Transform → Grab/Move

Kirjeldus

Üks kiiremaid asjade liigutamise viise kolmemõõtmelises ruumis on kiirvalik G. Selle nupu vajutamisel sisened "haaramise/liigutamise" teisendusrežiimi, kus saad objekte või andmeid vastavalt hiire kursori asukohale liigutada. Kursori ja manipuleeritava objekti vaheline kaugus ei oma tähtsust.

Valikud

LMB 

Kinnita liigutamine ja jäta objekt või andmed nende praegusesse asukohta ekraanil.

MMB 

Piira liigutamist X-, Y- või Z-teljega.

RMB  või Esc

Tühista liigutamine ja aseta objekt või andmed tagasi nende algsesse asukohta.

Pööramine (*Rotate*)

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: R

Menu: Object/Mesh/jne → Transform → Rotate

Kirjeldus

Kiirvaliku R vajutamisel sisened "pööramise" teisendusrežiimi, kus saad objekti või andmeid vastavalt hiire kursori asukohale pöörata. See režiim kasutab pööramise nurgana hiire kursori ja objekti keskpunkti vahelist nurka. Seega hiire liigutamine objektist/andmetest kaugemal lubab suuremat täpsust (st hiire liigutamine mõjutab pööramist vähem kui võrdväärne liikumine siis, kui hiir on objektile lähemal).

Sätted

LMB 

Kinnita pööramine ja jäta objekt või andmed nende praegusesse kaldenurka ekraanil.

MMB 

Piira pööramist X-, Y- või Z-teljega.

RMB  või Esc

Tühista pööramine ja aseta objekt või andmed tagasi nende algsesse kaldenurka.

R (juhtkuul)

R vajutamine juba pööramisrežiimis olles võimaldab vahetada üksikteljel pööramise (joondatud kas ekraani või mõne konkreetse teljega) ning kahel teljel ehk "juhtkuulina" (trackball) pööramise eelistust. "Juhtkuuli" režiimis mõjutab objekti pööramist nii hiire kursori X kui ka Y asukoht (samamoodi nagu vaate pööramisel juhtkuuli režiimis olles). Nii saad objekti kiirelt õige nurga alla pöörata, ilma et peaksid pööramiseks pidevalt vaatenurka muutma. Samas on see harjumatum kui lihtne pööramine.

Teisendamisrežiimides kasutatavate lisasätete mõistmiseks loe ülejäänud objektide kolmemõõtmelises ruumis manipuleerimise lõiku.

Mõõtkava (*Scale*)

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: S

Menu: Object/Mesh/jne → Transform → Rotate

Kirjeldus

Kiirvaliku S vajutamisel sisened "mõõtkava" teisendusrežiimi, kus saad objekti või andmeid vastavalt hiire kursori asukohale suurendada ja vähendada. Objekti/andmete mõõtkava suureneb, kui liigutad hiire kursorit keskpunktist eemale, ning väheneb, kui liigutad kursorit sellele lähemale. Kui hiire kursor liigub algselt küljelt üle keskpunkti vastasküljele, siis muutub mõõtkava edasi negatiivses suunas ning objekt/andmed näivad peegelpildis. Mõõtkava muutmise täpsus sõltub hiire kursori kaugusest objektist/andmetest mõõtkava muutmist alustades.

Valikud

LMB 

Kinnita mõõtkava muutmine ja jäta objekt või andmed nende praegusesse mõõtkava ekraanil.

MMB 

Piira mõõtkava muutmist X-, Y- või Z-teljega.

RMB  või Esc

Tühista mõõtkava muutmine ja aseta objekt või andmed tagasi nende algsesse mõõtkava.

Keerulisemad teisendused

Peegeldamine (*Mirror*)


Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: CtrlM

Menu: Object/Mesh/jne → Mirror

Kirjeldus

Objektide peegeldamine pöörab nad telgedel pahupidi sõltuvalt nende [keskpunktist](#) ja on seega võrdväärne mõõtkava muutmisega väärtuseni -1.

Vajuta esmalt CtrlM ja seejärel X, Y või Z, peegeldamaks objekti vastaval teljel. Teise võimalusena võid all hoida klahvi MMB  ja liigutada hiirt soovitava peegeldustelje suunas.

Teisendamisrežiimides kasutatavate lisasätete mõistmiseks loe ülejäänud objektide kolmemõõtmelises ruumis manipuleerimise lõiku.

Teisenduste piiramine

Selles peatükis seletame, kuidas määrata:

- teisenduse muutuse hulka
 - [Teisenduse täpsus](#)
 - [Numbriline teisendamine](#)
 - [Teisendamise sätted](#)
 - [Objekti teisenduse algseadistamine](#)
 - [Proportsionaalne muutmine](#)
- teisenduse suunda
 - [Manipulaatorid](#)
 - [Teisendussuunad](#)
 - [Telgede lukustamine](#)
- teisenduse keskkoha ([keskpunkt](#))
 - [Aktiivne objekt](#)
 - [Individaalsed keskpunktid](#)
 - [3D-kursor](#)
 - [Mediaanpunkt](#)
 - [Piirdkasti kese](#)
- teisendusele eeskujuks olevat objekti
 - [Teisenduste nakkumine](#)
 - [Võrega nakkumine](#)

Teisendamise sätted (*Transform Properties*)

Igasse objekti on salvestatud tema asukoht, suund ning mõõtkava väärtused. Neid saab vajadusel numbriliselt muuta, tühistada või kinnitada.

Teisendamissätete paneel

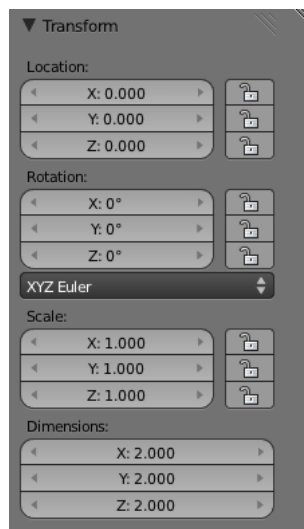
Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: N

Menu: Object » Transform Properties

Vaate omaduste paneeli View Properties teisendamissätete sektsioon Transform Properties lubab sul objektirežiimis Object vaadata ning käsitsi/numbriliselt muuta objekti asukohta, pööret või teisi omadusi. Muutmise režiimis Edit lubab see sul peamiselt sisestada tippude täpsed koordinaate või tippude grupi (k.a servad/küljed) mediaani asukohta. Kuna igat tüüpi objektil on muutmise režiimis Edit režiimis teisendamissätete paneelil Transform Properties erinevad valikud, vaata vastavaid kirjeldusi [modelleerimise peatükist](#).

Sätted objekti režiimis



Teisendamissätete paneel
Transform Properties paneel
objektirežiimis Object.

Location X, Location Y, Location Z (asukoht)
Objekti keskme asukoht globaalkoordinaadistikus.

Rotation X, Rotation Y, Rotation Z (pööre)
Objekti pööre, mis on seotud globaaltelgedega ja objekti enda keskmega.

Scale X, Scale Y, Scale Z (mõõtkava)
Objekti mõõtkava, mis on suhtes objekti keskmega ja esitatud kohalikes koordinaatides (st välja Scale X väärtus väljendab mõõtkava X-teljel). Igal objekti (kuubi, kera jne) mõõtkava on loomise hetkel igas suunas üks Blenderi ühik. Objekti suuremaks või väiksemaks tegemiseks muudad sa soovitud suunas selle mõõtmeid.

Dimensions X, Dimensions Y, Dimensions Z (suurus)
Objekti peamised mõõtmised (Blenderi ühikutes), mis on justkui joonlauaga ühest välisest servast teiseni mõõdetud. Mitme küljega pindadel on nende väljade sisuks piirdkasti mõõtmised (mis on joondatud kohalike telgedega — kujuta ette pappkasti, mis on täpselt piisavalt suur, et objekti endas hoida).

Kasuta seda paneeli, et muuta või lihtsalt näha objekti teisendamissätteid nagu asukoht, pööre ja/või mõõtkava. Need väljad muudavad objekti keset ning seejärel kõigi selle *tippude* ja külgede omadusi.



Märkus IPO-de kohta

Asukoha, pöörde ja mõõtkava väärtusi võib mõjutada ka IPO-võtmekaader, nii et kui objektiga on seotud IPO-võtmed, värskenda neid kindlasti pärast selles paneelis muudatuste tegemist, muidu lähevad sinu muudatused kaotsi, kui sa kaadreid uuesti muudad (IPO-võtmed tühistavad käsitsi pandud sätted).

Mõned väljad omavad ka lisafunktsioone või -võimalusi (nagu näiteks kerimisala). Seda tüüpi väljade muutmisel on lihtsam kasutada kombinatsiooni {⇧ Shift LMB tavalise LMB asemel. Pärast välja muutmist klõpsa hiirega sellest väljaspool või vajuta ↵ Enter, et muutused kinnitada. Vaateaken peegeldab muutusi automaatselt. Muutmise tühistamiseks vajuta Esc. Muudetava välja lisavõimaluste kirjeldusi loe [liidese](#) peatükist.

Teisendamissätete lukustamine

Asukoha, pöörde ja mõõtkava lukustamine laseb sul neid teisendamise väärtusi muuta ainult sätete paneelist. Kui lukk aktiveeritakse, on kõik teised teisendamisevõimalused keelatud. Kui sa näiteks lukustad asukoha välja Location X, siis ei saa sa enam kasutada hiirt objekti nihutamiseks globaalsel X-teljel. Aga välja Location X kasutades on nihutamine ka edaspidi võimalik. Lukustamist võib pidada karmiks keeluks, mida saab muuta ainult sellest paneelist.

Välja lukustamiseks vajuta selle kõrval olevat tabaluku ikooni. Väli on lukustamata, kui lukk näeb välja selline () ja lukustatud, kui lukk näeb välja selline ()

Teisenduste kinnitamine

Teisendusväärtuste kinnitamine põhimõtteliselt lähtestab objekti asukoha, pöörde või mõõtkava, kuid ei tee objektiga tegelikult midagi. Keskpunkt tõstetakse globaalkoordinaatide algpunkti ja teisendusväärtused muutuvad nulliks. Mõõtkava väärtused muutuvad uuesti üheks.

Teisenduse rakendamiseks vali objektimenüü Object alammenüü Apply või kasuta kiirvalikut CtrlA ning vali sealt vastav teisendus, mida rakendada.

Duplikaatide reaalseks tegemise valik Make Duplicates Real lingib duplikaadid lahti ja muudab neist igaühe eraldi andmeplokiks.

Teisenduste tühistamine

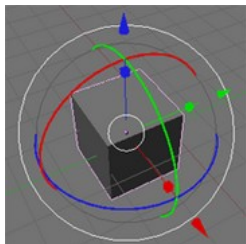
Teisenduste tühistamine lihtsalt lähtestab teisenduste väärtused. Objekti koha ja pöörde väärtused muudetakse 0-ks ja mõõtkava 1-ks.

- Clear Transform (tühista teisendus, AltG)
- Clear Rotation (tühista pööre, AltR)
- Clear Scale (tühista mõõtkava, AltS)
- Clear Origin (tühista keskpunkt, AltO)

Manipulaatorid (*Manipulators*)

Mode: Muutmisrežiim Edit ja objektirežiim Object


Hotkey: CtrlSpace



Erinevad manipulaatorid.

Kui kasutada tavalisi teisendamise käske (G nihutamiseks, R pööramiseks, S mõõtkava muutmiseks), töötavad nad ainult avatud vaatega paralleelselt.

Enamasti pead kasutama [telgede lukustamist](#):

- koheselt pärast G, R või S vajutamist võid sa vajutada X, Y või Z, et teisendamist mõne globaalse teljega piirata
- MMB  kasutamine teeb sama, kuid piirab liigutamise kursorile lähima teljega.

Sama tulemuse saavutad **manipulaatoreid** kasutades: nende kuju vihjab, mis tüüpi teisendust sooritatakse, ning nad võimaldavad liigutada, pöörata ja mõõtkava muuta mööda ükskõik millist telge ning 3D-akna igas avas (vaatenurgas).

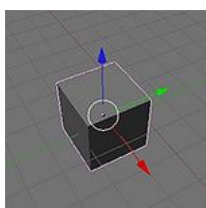
Iga teisenduskäsu jaoks on erinev manipulaator. Igaüht saab vaadata eraldi või samaaegselt teistega.



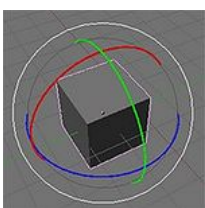
Manipulaatori päis.

Manipulaatoriteni pääseb läbi 3D-vaate akna päise:

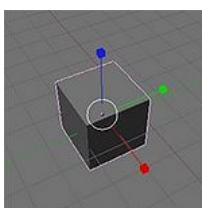
- Teljed: lülitab manipulaatorite kasutamist.
- Nool: nihutamine/asukoht.
- Kaar: pööramine.
- Ruut: mõõtkava.
- Teisendussuuna menüü: selle kaudu saad valida teisendussuuna.



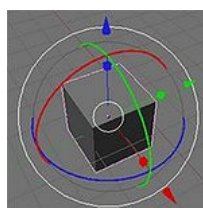
Nihuta.





Pööra.








Muuda mõõtkava.



Kõik manipulaatorid koos.

Vajutades manipulaatorite ikoonidele (nool, kaar, ruut) kombinatsiooniga  Shift LMB , valitakse korraga rohkem kui üks manipulaator.

Manipulaatorite kasutamine

- Nupu Ctrl all hoidmine piirab tegevust lühikeste sammudega (rohkem infot [siin](#)).
- Nupu  Shift allhoidmine ja hiire klahviga LMB  mõnele manipulaatori sangale vajutamine piirab manipulaatori tegevust sinu poolt klikimata jäänud teise kahe teljega (pärast hiirega klõpsamist võid nupu  Shift lahti lasta — loe lähemalt ka [tasapinnalisest telgede lukustamisest](#))
- LMB  vajutamine valgel ringil (pööramise manipulaatori ümber asuv suurim ring) teeb sama, mida kiirvaliku R vajutamine.
- LMB  vajutamine hallil ringil (pöörlemise manipulaatori ümber asuv sisemine ring) teeb sama, mida kiirvaliku R vajutamine kaks korda järjest (ehk juhtkuulina (trackball) pööramine, vaata [siit](#)).

Manipulaatori sätted

Manipulaatori sätted.

Manipulaatori sätted (nt selle suurus) leiad kasutaja eelistuste akna (User Preferences) vaate ja juhtimisseadmete osast (View & Controls).

- Size (suurus): manipulaatori enda suurus 10-pikslistes ühikutes.
- Handle (sang): manipulaatori sangade suurus protsendina manipulaatori raadiusest (Size/2).
- Hot spot (haarderaadius): sangade ümber jääva haarderaadius suurus (pikslites).

Teisendussuuna valimine

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: AltSpace

Vaikimisi valitud (globaalne) teisendamise manipulaator on väga kasulik, kuid mõningatel juhtudel sellest ei piisa (näiteks pööratud objekti suuruse muutmine lähtuvalt pöördenurga teljest). Õnneks saab Blender teisendamise manipulaatori suunda muuta.

Täpsemalt loe teisendussuundade kohta [sellelt lehelt](#).

Teisendussuunad

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: AltSpace

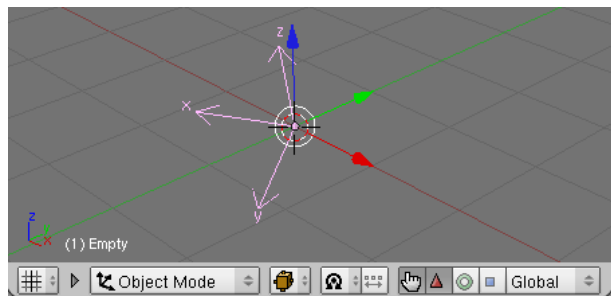


Teisendussuuna valimise menüü.

Teisest suunda saab valida, vajutades AltSpace või läbi 3D-vaate päises asuva suunamenüü Orientation. Seda suunda saab kasutada teisenduse piirajana (nagu [telgede lukustamistki](#)).

Valikud

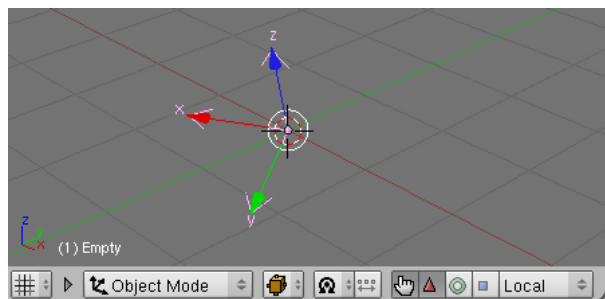
Altpoolt leiad erinevad teisendussuunade liigid. Võrdle igal pildil manipulaatoritelgede asukohta (objekti peal olevad värvilised teljed), mis näitavad teisendussuunda ennast, globaalsete (3D-akna alumises vasakus nurgas) ja lokaalsete (kuna tegemist on tühiobjektiga, näidatakse ainult objekti lokaalseid telgi) telgede asukohaga.



Globaalne (Global).

Global (globaalne)

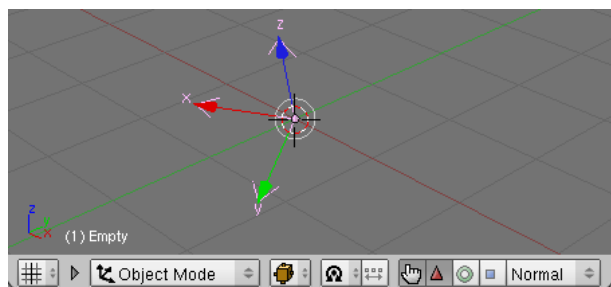
Manipulaator ühtib globaalsete telgedega.



Lokaalne (Local).

Local (lokaalne)

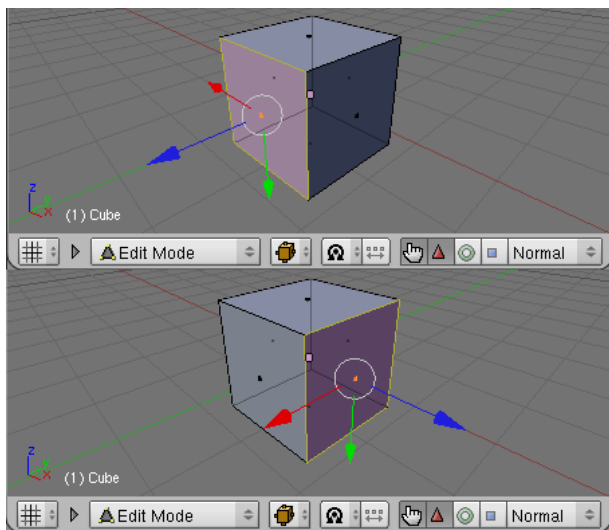
Manipulaator ühtib objekti telgedega.



Normaal (Normal).

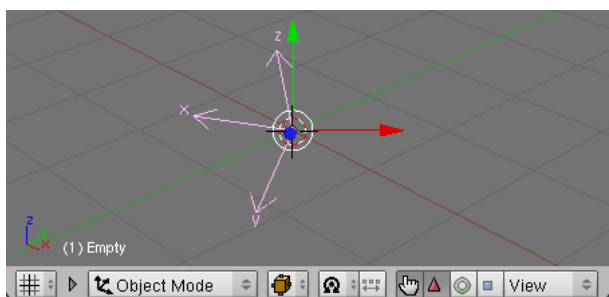
Normal (normaal)

Manipulaatori Z-telj kattub valitud objekti normaali vektoriga. Tühiobjekti puhul pole see eriti kasulik, nii et vaata allolevat näidet.



Kuubi näide.

Parem näide, mis kujutab küljenormaalidega objekti ning külgede valimist muutmisrežiimis Edit.



Vaade (View).

View (vaade)

Manipulaator ühtib 3D-vaatega, Y → üleval/all, X → vasakul/paremal, Z → sinu suunas/sinust eemale.

Kohandatud suunad

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: ☆ ShiftCtrlC

Saad luua kohandatud teisendussuundasid, mis kasutavad objekti või võre elemente. Kohandatud teisendussuund, mis on loodud objektidest, kasutavad objekti kohaliku suunda, ning need, mis on loodud võre elementidest (tipud, servad, küljed), kasutavad valiku normaali suunda.

Uuele suunale tuleb panna ka nimi.

Märkus

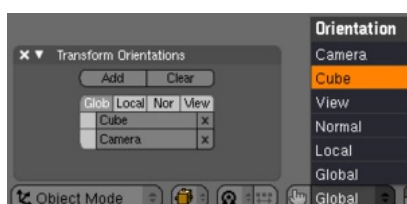
Kui uusi suundasid lisades vastab sisestatud nimi juba mõnele olemasolevale kohandatud teisendussuunale, siis asendab uus suund vana.

Teisendussuundade paneel

Teisendussuundade paneeli (Transform Orientations), mis asub vaate sätete paneelis (View Properties Panel), saab kasutada teisendussuundade haldamiseks: aktiivse suuna valimiseks, kohandatud suundade lisamiseks ja eemaldamiseks ning kõigi kohandatud suundade kustutamiseks.



Teisendussuundade paneel (Transform Orientations).



Kohandatud suuna valimine.

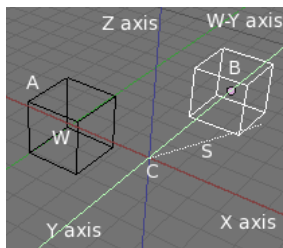


Teisendussuundade paneel koos valitud kohandatud suunaga.

Telgede lukustamine (*Axis Locking*)

Mode: Muutmisrežiim Edit ja objektirežiim Object (nihuta, pööra, teisenda mõõtkava, eenda)

Hotkey: X, Y, Z



Globaalteljed.

Liigutusi (nihutamist, mõõtkava muutmist, pöörast ja väljundamist muutmisrežiimis Edit) saab piirata teljega, mis on paralleelne mõne [teisendusteljega](#) ja tsentreeritud objekti algsele asukohale.

Pildil (*Globaalsed teljed*) on kuubi "B" algne asukoht maailmas tähistatud tähega "C". Globaalse koordinaatteljestiku keskpunkt on tähistatud tähega "W" ja Z-telj ei ole nähtav.

Piirates teisendamist ühe globaalse teljega, viid sa selle reaalselt läbi ühes dimensioonis.

Kõik Blenderi teljed kasutavad XYZ <-> RGB (*red, green, blue* — punane, roheline, sinine) analoogial põhinevat värvisüsteemi:

- **X-telj on punane** (tähistusega "X axis").
- **Y-telj on roheline** (tähistusega "W-Y axis" ehk "maailma Y telg" ("world-Y axis")).
- **Z-telj on sinine** (tähistusega "Z axis").

Nende toon on heledam või tumedam vastavalt sellele, kas nad on "lukustatud" või mitte.

Näiteks kui liigutamine on piiratud Y-teljega, kujutatakse telge helerohelisena; näitejoonisel sildiga "Y axis".

Järgnevalt kirjeldame selle funktsionaalsuse erinevaid sätteid ja tüüpe.

Telgede lukustamise viisid

Telgede lukustamine (*Axis locking*)

Lukustab teisendamise ainult ühele teljele (ehk siis keelab kahel teljel teisendamise).

Objekt (või tipp, kontrollpunkt või misiganes) saab liikuda, mõõtkava muuta või pöörata ainult ühes suunas, mööda sirget joont või selle joone ümber.

Tasapinnaline lukustamine (*Plane locking*)

Lukustab *kaks* telge (ehk keelab teisendamise ainult ühel teljel) ja loob seeläbi tasapinna, mida mööda saab elementi vabalt liigutada või selle mõõtkava muuta.

Tasapinnaline lukustamine puudutab ainult nihutamist ja mõõtkava muutmist.

Pane tähele, et pööramise jaoks on nii telgede kui ka tasapinnalises lukustamisel sama toime, sest pööramine toimub alati ümber ühe telje.

Juhtkuuliga pööramine

Pea samuti meeles, et juhtkuuli (*trackball*) stiilis pööramisi (RR) ei saa üldse lukustada!

Telgede lukustamise režiimid

Saad teisendamist teljepõhiselt lukustada erinevates ruumides. Vaikimisi lukustad ainult maailma (globaalsed) teljed. Kui kasutad klaviatuuri telgede valimise meetodit (vaata altpoolt), on sul lisaks võimalik valida parajasti kehtiva [teisendussuuna](#) telgede vahel (kui see ei ole parajasti globaalne (Global), mispuhul on teine valik alati lokaalne (Local)).

Kuidas telgesid lukustada

Liikumist saab piirata kahel moel: kasutades hiirt või klaviatuuri.

Kasutades klaviatuuri

Saad teisenduse piirata ühe teljega vajutades X, Y või Z ning tasapinnaliselt vajutades ⇧ ShiftX, ⇧ ShiftY või ⇧ ShiftZ.

Üksik nupu vajutus piirab liikumise vastavale *globaalsele* teljele (pilt *Globaalne piirang*). Teine *sama* nupu vajutus parajasti valitud teisendussuunale (välja arvatud juhul, kui see on määratud globaalseks (Global): sel juhul kasutatakse lokaalset (Local) telge). Ja kolmas sama nupu vajutus tühistab piirajad (pilt *Piiramata*). Lukustatud telgi näidatakse selguse huvides heledamalt. Pildid (*Lokaalne piirang*) ja (*Globaalne piirang*) näitavad mõlemad X-telje lukustamist peale nupu X vajutamist.

NB! Telgede lukustamine ei takista [klaviatuuriga](#) täpsete teisendamise väärtuste sisestamist.

Dx: -0.3125 Dy: -0.1831 Dz: 0.1547

Piiramata.


D: -0.5466 along global X

Globaalne piirang.

D: -0.8295 along local X

Lokaalne piirang.

Kasutades hiirt

Lukustamiseks või piiramaks liikumist hiirega, sisene nihutamise, mõõtkava muutmise või pööramise režiimi (Grab/Scale/Rotate) ja vii vastav toiming läbi, hoides all nuppu MMB . Teisendusrežiimis olles saad kasutada telje eelvalimiseks [viipesüsteemi](#), liigutades hiirt enam-vähem vastava maailma teljega samas suunas ning seejärel vajutades korra hiireklahvi MMB . Näiteks kui liigutad hiirt näiliselt mööda X-telge ning vajutad seejärel klahvi MMB , piiratakse objekti liikumine X-teljega.

Teise võimalusena võid piirava telje valida nihutamisrežiimis (Grab), hoides klahvi MMB all ja hiirt lohistades. Kõik kolm telge muutuvad nähtavaks ning objekti algsest asukohast (tähistatud tähega "c") väljub abijoon. Selle joone tähiseks on tähega "s" tähistatud valge punktiir. Kui joon läheneb mõnele teljele, muutub see telg heledamaks ning objekt nakkub selle külge. Antud näidises on abijoon Y-telje lähedal ja tähega "b" tähistatud kuup on selle külge nakkunud.

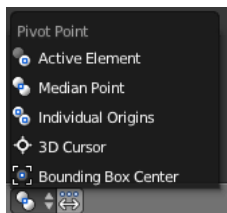
Seda ei ole lihtne seletada, seega proovi kohe järele: märkad, et tegemist on üsna intuitiivse süsteemiga! Pea meeles, et hiirega saad sa valida ainult telge ning globaalse ja parajasti valitud teisendussuuna vahel valimine pole võimalik. Samuti ei saa sa liikuda "normaalse" ja "tasapinnalise" telgede lukustamise viisi vahel... Selleks pead kasutama klaviatuuri.

Keskpunkt (*Pivot Point*)

Keskpunkti valimine

Mode: Muutmisrežiim Edit ja objektirežiim Object

Menu: 3D-vaate päises asuv rippmenüü



Keskpunkti režiimid.

Keskpunkt on see ruumipunkt, ümber mille kõik pööramised, mõõtkava muutmised ja peegeldamised aset leiavad. Võid valida viie keskpunkti režiimi vahel, mida saab 3D-ala päises olevast rippmenüüst muuta, nagu on näha ka pildilt (*Keskpunkti režiimid*). Sinu ülesandeks on valida käsiloleva töö jaoks efektiivseim režiim ning paigutada keskpunkt võimalikult täpselt.

- [Aktiivne objekt](#)
- [Mediaanpunkt](#)
- [Individaalsed keskpunktid](#)
- [3D-kursor](#)
- [Piirdkasti kese](#)

Pane tähele, et kuigi ülal olevad näited kasutavad võresid, kehtivad samad reeglid aga ka teistele tüüpidele (köverad, pinnad...).

Objektide keskpunktid

Igal objektil on oma keskpunkt. Sellest punktist sõltub, kus objekt kolmemõõtmelises ruumis asetseb. Valitud objekti puhul ilmub ekraanile väike objekti keskpunkti märkiv täpp. Keskpunkti asukoht on oluline objekti pööramisel ja mõõtkava muutmisel.

Objektide keskpunkti liigutamine

Objektide keskpunkte saab objekti menüü (*Object*) alammenüü (*Transform*) abil liigutada erinevatesse asukohtadesse:

- 3D-kursori asukohta (*3D Cursor Location*)

Liigutab keskpunkti 3D-kursori asukohta. Uuri lehelt [3D-vaate kasutamine](#) lähemalt 3D-kursori kasutamise kohta.

- Geomeetria mediaanpunkti (*Median Point of Geometry*)

Liigutab keskpunkti objekti komponentide arvutuslikku keskmesse.

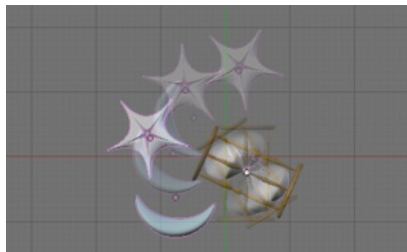
Aktiivne objekt (*Active Object*)

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: Alt.

Unusta see nimi: see ei ole piiratud ainult objektidega. Muutmisrežiimis Edit võib *aktiivseks* elemendiks olla ka tipp, serv või külg.

Objekti režiimis



Ümber aktiivse objekti pööramine.

Objektirežiimis Object toimub see suhteliselt lihtsalt: pööramised ja mõõtkava muutmised toimuvad ümber aktiivse objekti keskpunkti. Seda illustreerib pilt (*Ümber aktiivse objekti pööramine*), kus aktiivse objekti (liivakell) keskpunkt on ainus, mis jääb omale kohale.

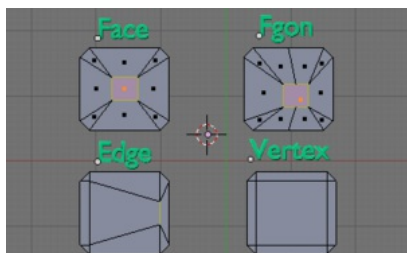
Muutmisrežiimis

Näiliselt on muutmisrežiimis Edit aktiivse elemendi kasutamine keskpunktina väga keeruline, kuid tegelikult sõltuvad kõik erinevad võimalused ainult mõnest üksikust reeglist:

- Keskpunkt on alati aktiivse(te) elemendi/elementide mediaan.
- Teisendamised toimuvad aktiivse(te) elemendi/elementide **tippude** teisendamise teel. Kui valimata element jagab ühte või enam tippu valitud elemendiga, siis teisendatakse teataval määral ka valimata elementi.

Uurime lähemalt järgnevat näidiseid ning näeme, et igaühe puhul on need mõlemad reeglid kaetud.

Üksikvalik



Muutmisrežiim Edit vaid ühe valitud elemendiga.

Kui üksainus element on valitud, muutub see automaatselt aktiivseks. Pildil (*Muutmisrežiim Edit vaid ühe valitud elemendiga*) näed sa, et kui valitud elementi teisendatakse, liiguvad selle tipud, mille tagajärjeks on see, et ka naabruses olevad elemendid, mis jagavad aktiivse elemendiga mõnda tippu, saavad natuke teisendatud.

Vaatame igat erinevat juhtu:

- *Külgede* keskpunkt asub seal, kuhu ilmub nende valikutäpp, ehk siis kohas, mis on nende tippude mediaan.
- *Fgonid* käituvad samamoodi, kuid pane tähele, et valiku täpp võib erinevalt külgedest olla natuke teises kohas.
- *Servade* keskpunkt on nende keskel, sest seal asub serva mediaan.
- Üksikul *tippul* ei ole dimensioone ning seega ei saa ta ka oma teisendamist näidata (välja arvatud nihutamist, kuid seda keskpunkt ei mõjuta...).

Mitmikvalik



Muutmisrežiim Edit mitmikvalikuga.

Kui valitud on mitu elementi, siis teisendatakse neid kõiki.

Keskpunktid jäävad samale kohale nagu eelmistes näidetes, välja arvatud Fgonite puhul. Pildil (*Muutmisrežiim Edit mitmikvalikuga*) on valitud elementide mõõtkava vähendatud:

- *Külgede* puhul toimub teisendus aktiivse külje keskpunkti ümber.
- *Fgonid* käituvad samamoodi nagu küljed.
- *Servad* käituvad samuti nagu enne ning nende keskpunkt asub nende mediaanis.
- Seekord on *tipud* nähtavad: nende keskpunkt asub aktiivse tipu asukohas. Kõiki teisi tippe teisendatakse sellest sõltuvalt.

Kordame — nagu oleme näinud, kehtib kaks reeglit:

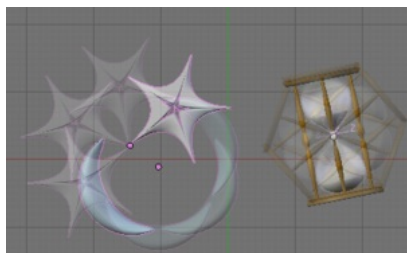
- Keskpunkt asub aktiivse elemendi tipu või tippude mediaanil.
- Teisendatakse ainult valitud tippe (hoolimata sellest, kas nad on valitud otse või mõne suurema elemendi (nt külje) osana).

Individaalne keskpunkt (*Individual Center*)

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: Ctrl.

Objekti režiimis



Ümber individuaalsete keskpunktide pööramine.

Pildil (*Ümber individuaalsete keskpunktide pööramine*) on iga objekti keskpunkt jäänud samasse asukohta ning igaüks neist pöörleb ümber selle.

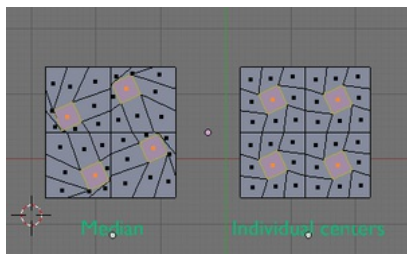
Objektide keskpunkti asukohta määramine laseb sul animatsiooni üle suuremat kontrolli omada. Uurime natuke pilti (*Ümber individuaalsete keskpunktide pööramine*):

- Liivakella keskpunkt langeb kõigi selle osade mediaani.
- Paigutades tähe keskpunkti ühe selle teraviku tippu, on meil võimalik seda ringis pöörata, kasutades pööramise IPO-sid (Rot) ning seetõttu on interpolatsioon täpsem kui samas asukohas oleva 3D-kursori kasutamisel.
- Poolkuu keskpunkt on täielikult meie poolt nähtavast objektist väljas. Pead mõistma, et keskpunkt tähistab seda täpset kohta, kus **objekt asub**; mida meie ekraanil näeme, on vaid kirjeldus sellest, millest objekt koosneb (tipud, värvid, asjad), ning seetõttu võib keskpunkt olla täiesti objekti keskkohast eemal. Nagu ta siinse poolkuu puhul ka on.

Muutmisrežiimis

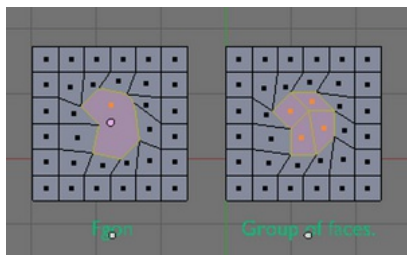
Kui kasutad valimiseks tippude või servade režiimi, on valitud tippude või servade keskpunktiks nende asukohtade mediaan. Täpsemalt loe osast, mis kirjeldab [mediaanpunkti](#) kasutamist keskpunktina.

Kui aga oled valinud külgede režiimi, on võimalik kasutada keskpunktina nende külgede keskpunkte.



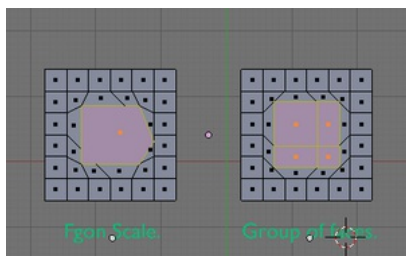
Mitme külje individuaalne pööramine.

Igat külge on eraldi pöörata võimalik ainult külgede valimise režiimis. Ainult külgi, mis üksteisega kokku ei puutu, on võimalik sedasi ilma moonutuseta teisendada. Kui teisendad üksikuid külgi sedasi, saad kasutada selleks proportsionaalse muutmise tööriista (PET).



Fgonide pööramine iga üksiku elemendi keset keskpunktina kasutades.

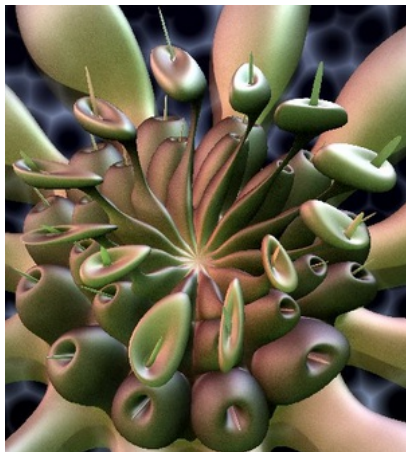
Küljed, mis omavahel kokku puutuvad (isegi kui nad on fgoni sees), moonutuvad, kui neid individuaalseid keskmey keskpunktina kasutades pöörata.



Probleemid fgonide ja külgede grupi mõõtkava teisendamisel.

Fgone ja külgede gruppide mõõtkava saab teisendada nii, et nende väline perimeeter ei moonutu. Samas peaksid sa silmas pidama, et üksikuid külgi nende sees ei teisendata ühtlaselt.

Kõiki neid moonutusi saab ära hoida, kui sa ei kasuta külgede valimise režiimi, kuid seeläbi muutub võimatuks enam kui ühe külje või külgede grupi samaaegne muutmine.



Külgede modelleerimine, kasutades individuaalseid keskmeid keskpunktina.

Kui sa lõpuks mõistad selle meetodi piire ja piiranguid, võib see tööriist hoida kokku kõvasti aega ning viia sind unikaalsete kujunditeni. See "meriroos" modelleeriti umbes kümne minutiga 12 küljega silindrist, kasutades korduvalt tegevuste järjekorda: üksiku külje eendamine (*extrusion*), selle mõõtkava teisendamine, kasutades *keskpunktina mediaani*, ning saadud külgede mõõtkava ja pöörde teisendamine, kasutades *keskpunktina individuaalseid keskpunkte*.

3D-kursor (*3D Cursor*)

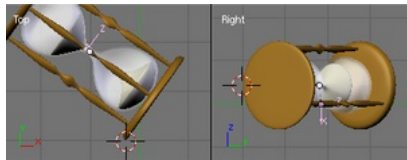
Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: .

3D-kursor on kõige lihtsam ja intuitiivsem keskpunkt, millega saad tulemusi täielikult juhtida. Seda saab kokku võtta järgnevalt: aseta kohale ja teisenda.

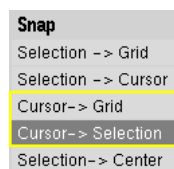
3D-kursori asetamine

3D-kursori asetamiseks on mõned erinevad meetodid.



3D-kursori asetamine kahe ortogonaalse vaate abil.

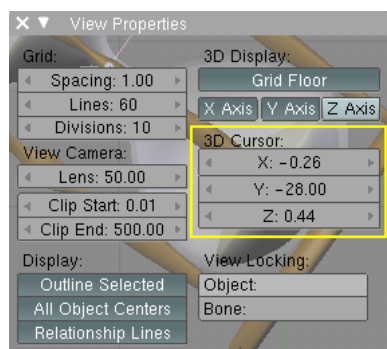
- Kasutades 3D-alas hiireklahvi LMB . Täpsemaks tulemuseks peaksid sa kasutama kahte paralleelprojektsioonis ristuvat 3D-vaadet, st kombinatsiooni ülalt (7 NumPad), eest (1 NumPad) ja küljelt (3 NumPad). Sedasi saad sa ühes vaates kontrollida asetust kahe telje suhtes ning teises vaates asetuse sügavust.



Nakkumise menüü Snap.

- Nakkumise kasutamine:
 - Valiku ⇧ Shift → Cursor -> Grid abil saad saata 3D-kursori lähimasse **nähtavasse** ruudustiku punkti.
 - Valiku ⇧ Shift → Cursor -> Active abil saad saata 3D-kursori **aktiivse** (viimati valitud) objekti, serva või külje keskohta või **aktiivsesse** tippu.
 - Valiku ⇧ Shift → Cursor -> Selection abil saad saata 3D-kursori:
 - objekti keskohta
 - tippu.
 - Kui valitud on rohkem kui üks element ning kasutatakse piirdkasti (Bounding Box Center) keskpunkti, siis saadab valik ⇧ ShiftS 3D-kursori:
 - Objektirežiimis objektide keskmeid ümbritseva **piirdkasti keskele**
 - Muutmisrežiimis valitud tippu ümbritseva **piirdkasti keskele** (seda isegi servade või külgede režiimis, sest arvesse võetakse kaudselt valitud tippu).
 - Kui on valitud mediaanpunkti (Median Point) keskpunkt, saadab valik ⇧ ShiftS 3D-kursori:
 - Objektirežiimis objektide keskpunktide mediaani
 - Muutmisrežiimis valitud tippude mediaanpunkti.

Palju võimalusi *on* palju võimsust!



Vaate valikute paneel View Properties.

- Saad kasutada vaate valikute paneeli (View Properties...) kirjet menüüs View (3D-vaate päisest) ning sisestada 3D-kursori asukoha ilmuvasse paneeli numbriliselt 3D-kursori sektsioonis 3D Cursor.

Teisendamine

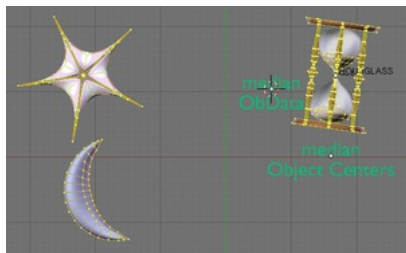
Seejärel ei tulegi muud teha, kui valida keskpunktina 3D-kursor ning pöörata, mõõtkava muuta või peegeldada.

Mediaanpunkt (*Median Point*) keskpunktina

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: Ctrl,

“Mediaanpunkti” võib kujutada raskuskeskmeks: eeldades, et igal valitud elemendil on sama mass, istuks mediaanpunkt raskuskeskmes ehk siis kohas, kus valitu oleks tasakaalus. See näide on aga järgnevat arvestades suhteliselt halb analoogia. Samas aitab mediaanpunkti ennustamisvõime kaasa stseeni planeerimisele.



Objekti kesete ja objekti andmete (*ObData*) mediaanpunkt.

Objektirežiimis

Objektide puhul võetakse arvesse ainult objekti kese. Ja mis veelgi olulisem — iga objekti keskme massi peetakse samaks. See võib viia väga ebaintuiivsete tulemusteni. Pildil (*Objekti keskmete ja objekti andmete (ObData) mediaanpunkt*) näeme, et objektide mediaan asub objektide keskmest kõvasti eemal. See tuleneb sellest, et kuu ja tähe objekti andmed (ObData ehk geomeetria) asuvad nende objektide keskpunktist väga kaugel.

Muutmisrežiimis

Ikka veel pilti (*Objekti keskmete ja objekti andmete (ObData) mediaanpunkt*) vaadates näeme, et ka objekti andmete (ObData) mediaanpunkt on liivakellale üllatavalt lähedal: see tuleneb sellest, et liivakellal on rohkem tippe (611) kui kuul (81) ja tähel (130). Blender peab kõigi tippude massi võrdväärseks.

Teisendamine

Kui nimekirjast on valitud mediaanpunkt (Median Point), siis teisendusmanipulaator (kui see on sisse lülitatud) nakkub koheselt sellega ning annab ülihea visuaalse vihje: kõik pöörlemised, mõõtkava muutmised ja peegeldamised saavad toimuma ümber selle punkti.

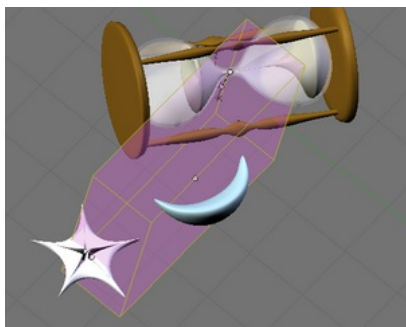
Piirdkasti kese (*Bounding Box Center*) keskpunktina

Mode: Muutmisrežiim Edit ja objektirežiim Object

Hotkey: ,

Piirdkast on ristkülikukujuline kast, mis on võimalikult tihedalt määsitud ümber valiku. See asetseb paralleelselt maailma telgedega. Selles režiimis asub keskpunkt piirdkasti keskmes.

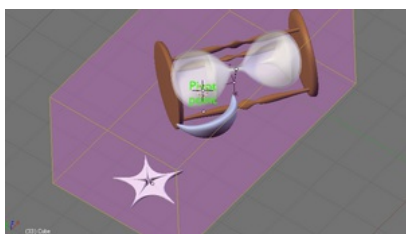
Objekti režiimis



Piirdkast objektirežiimis.

Objektirežiimis on piirdkast mähitud ümber objektide keskmete ja ei võta seega arvesse objekti andmeid (*ObData* ehk geomeetria).

Muutmisrežiimis



Piirdkast muutmisrežiimis.

Sellel juhul on just objekti andmed (*ObData*) need, mis on piirdkasti sees, sest valitud on kõik tipud. Muutmisrežiimis ei arvesta piirdkast objektide keskmatega, vaid ainult valitud tippudega.

Nakkumine (*Snap*)

Hotkey: ⇧ ShiftS

Menu: Object » Snap

Selection to Grid (valik ruudustikuga)

Valitud objekt(id) nakkub/nakkuvad lähima ruudustiku punktiga.

Selection to Cursor (valik kursoriga)

Valitud objekt(id) nakkub/nakkuvad kursori asukohaga.

Cursor to Selected (kursor valituga)

Liigutab kursori valitud objekt(ide) keskohta.

Cursor to Center (kursor keskkohaga)

Liigutab kursori ruudustiku keskohta.

Cursor to Grid (kursor ruudustikuga)

Liigutab kursori lähimasse ruudustiku punkti.

Cursor to Active (kursor aktiivsega)

Liigutab kursori aktiivse objekti keskohta.

Teisenduse nakkumine

Objektide nakkumiseks erinevate stseeni osadega tuleb kasutada 3D-vaate päisenuppude hulgas olevat magneti ikooni. Objekti režiimis saad sellega valitud komponente sundida liigutamisel teiste objektide külge kleepuma.

Nakkumisrežiimid

Nakkumise element (*Snap Element*)

Increment (samm)

Nakkub ruudustiku punktide külge. Ortograafilises vaates sõltub nakkumise sammu suurus suurenduse astmest.

Vertex (tipp)

Nakkub võreobjektide tippude külge.

Edge (serv)

Nakkub võreobjektide servade külge.

Face (külj)

Nakkub võreobjektide küljepindade külge. See on kasulik topoloogia muutmisel.

Volume (maht)

Nakkub objektide mahtude piirkondade külge (esimese hiire kursori alt leitud mahu külge). Erinevalt teistest režiimidest muudab see valik (kehtivas vaates) teisendatava elemendi sügavust (st vaate Z-koordinaate) — element asetatakse hiirega valitud "mahu keskmesse". Valides nakkumise sihtmärgi menüüst paremale ilmuva nupu, peetakse sihtobjekte mahu keskme arvutamisel üheks suureks objektiks.

Nakkumise sihtmärk (*Snap Target*)

Nakkumise sihtmärgi valik muutub aktiivseks, kui nakkumise elemendi menüüs on valitud tipp (Vertex), serv (Edge), külj (Face) või maht (Volume). Need valikud määravad, milline objekti osa sihtobjekti külge nakkub.

Closest (lähim)

liiguta valiku lähim punkt sihtmärgi juurde.

Median (mediaan)

liiguta valiku mediaanpunkt sihtmärgi juurde.

Center (keske)

liiguta praegune teisenduskese sihtmärgi juurde. Seda saab kasutada koos 3D-kursoriga, et nakkuda teatava nihkega.

Active (aktiivne)

liiguta aktiivne element (muutmisrežiimis tipp, objekti režiimis objekt) sihtmärgi juurde.

Pöörde joondamine

Joonda objekti Z-telg sihtelemendi normaaliga, vajutades nakkumise sihtmärgi menüü kõrvale objekti režiimis ilmuvat nuppu.

Nakkudes servadega, arvutatakse normaal välja kahe tipu normaalidest. Muudel juhtudel kasutatakse otse külje ja tipu normaale.

See töötab ainult objekti režiimis tehtud nihutamise puhul.

Proportsionaalne muutmine

Objektirežiim

Mode: Objektirežiim Object

Hotkey: O

Menu: Mesh » Proportional Editing

Proportsionaalset muutmist kasutatakse enamasti muutmisrežiimis Edit, kuid seda saab kasutada ka objektirežiimis Edit.

Objekti režiimis tegeleb tööriist tervete objektide ja mitte üksikute võre osadega.

Muutmisrežiim

Mode: Muutmisrežiim Edit

Hotkey: O / AltO / ⇧ ShiftO

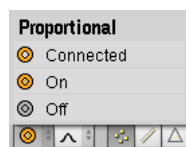
Menu: Mesh » Proportional Editing

Tihedate võreobjektide töötlemisel on väga raske üksikuid punkte muuta, ilma et mudeli pinnale tekiksid muhud või voldid. Sellisel puhul on abiks mõjualaga muutmise tööriist. Selle sisselülitamisel liiguvad ühe tipu liigutamisel mõnevõrra kaasa ka naabruses olevad tipud. Mida lähemal on valitud tipule naabertipp, seda rohkem too sellega kaasa liigub.

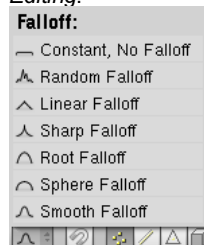
Voolimine (*Sculpting*)

Blenderil on olemas ([voolimisrežiim \(Sculpt Mode\)](#)). Voolimisrežiimis on mõjualaga muutmiseks rida tööriistu, mida saab kasutada ilma konkreetseid tippe valimisse võtmata.

Valikud



Mõjualaga muutmise nupp
Proportional Editing.



Mõjuala kõvera menüü *Falloff*.

Mõjualaga muutmise menüü ilmub nähtavale 3D-vaate päises, kui ollakse muutmisrežiimis.

Off (väljas, O)

Mõjualaga muutmine on välja lülitatud. Toimub ainult märgitud tippude mõjutamine.

On (sees, O või AltO)

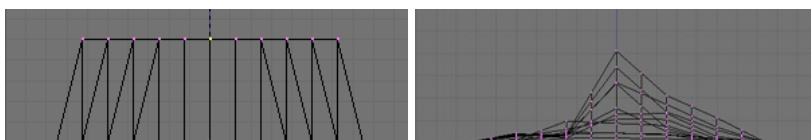
Mõjutatakse ka naabruses olevaid tippe, vastavalt määratud raadiusele.

Connected (ühendatud, AltO)

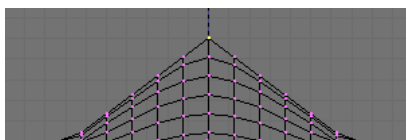
Kui ühe objekti sees on mitu omavahel ühendamata võret, siis mõjualaga muutmine mõjutab vaid märgitud tippudega võret. Kui ühendustee kahe tipu vahel on pikk, siis samuti ei toimu mõjutamist. Näiteks käe ühe sõrmeotsa muutmisel naabersõrm ei muutu. Kui on valitud Connected (ühendatud), muutub nupp keskoast halliks.

Falloff (kõver)

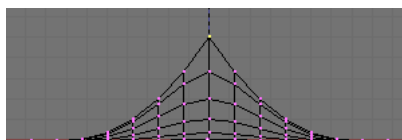
Mõjuala kõverat saab muuta Mesh → Proportional Falloff alammenüüst, tööriistaribalt Falloff menüüst (vt pilti *Mõjuala kõvera menüü Falloff*) või kiirvaliku ⇧ ShiftO abil erinevate võimaluste vahel liikudes.



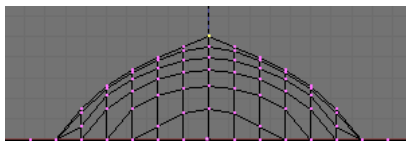
Constant, No Falloff - kõiki mõjualas olevaid tippe mõjutatakse võrdselt. Random Falloff - mõjuala tugevuse varieeruvus on juhuslik.



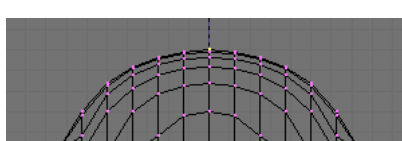
Linear Falloff - mõjuala mõju vähenemine on lineaarne.



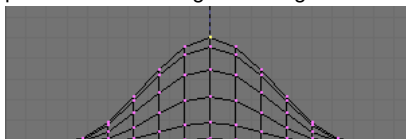
Sharp Falloff - mõjuala mõju vähenemise kõver on terav.



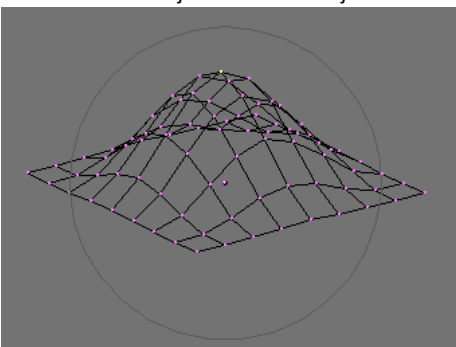
Root Falloff - mõjuala tugevus väheneb pöördvõrdeliselt kauguse ruuduga.



Sphere Falloff - kerakujuline mõjuala.



Smooth Falloff - sujuvalt kahanev mõjuala.



Mõjuala märkiv ringjoon.

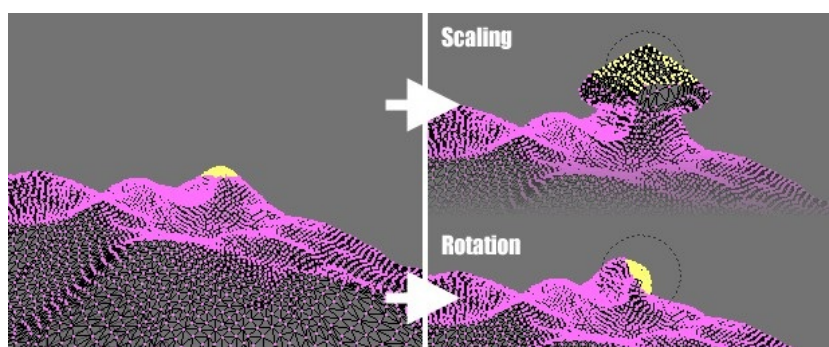
Mõjuala piir ja tugevus (*Influence*)

Mõjuala suurust saad muuta hiire rullikuga (WheelUp / WheelDown) või klahvidega PageUp/PageDown. Raadiuse suurendamisel suureneb vastavalt ka ümbritsev mõjuala.

Näited

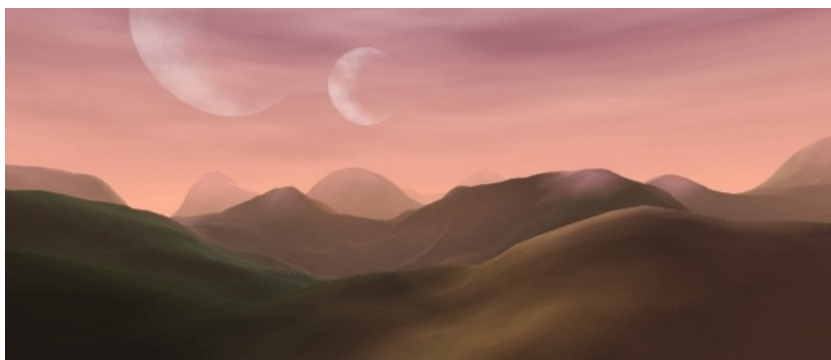
Lüliti eestvaatesse (1 NumPad) ja aktiveeri liigutamise tööriist, vajutades klaviatuuril G. Liigutades punkti ülespoole, on näha, et ka naabruses olevad tipud liiguvad osaliselt kaasa. Kui oled tulemusega rahul, klõpsa LMB hiireklahvi. Kui sa aga tulemusega rahul ei ole ja tahad endist seisu taastada, vajuta vasema hiireklahvi asemel Esc klahvi.

Mõjualaga muutmise korral saab suurendamise ja pööramisega igasugu ägedaid efekte luua. Selliseid, nagu on näha pildil "Mõjuala muutmise abil modelleeritud maastik".



Mõjuala muutmise abil modelleeritud maastik.

Toodud tehnikate kombineerimisel koos tippude maalimisega (*vertex painting*) saab vormida fantastilisi maastikke. Pilt *Lõplik renderdatud maastik* illustreerib proportsionaalse muutmise võimalusi pärast seda, kui on paika pandud tekstuurid ja valgustus.



Lõplik renderdatud maastik.

Sissejuhatus

See lehekülj ja selle alamleheküljed põhinevad [versiooni 2.48 rasvakriidi väljalaskemärkustel](#).

Võimalusest vaba käega vaateakende sisse ja/või peale visandite ja märkmete joonistamiseks on palju kasu koostöö ajal suhtlemises ja planeerimises. Seda võib seostada traditsioonilise kahemõõtmelise pliiatsi-ja-paberi tööprotsessiga, kus esmaseid "juhendusvisandeid" kasutati tihti planeerimiseks ja ka kiireks ideede edasiandmiseks. Sageli on kasulik sirgelda otse töös oleva taiese peale ning mitte kuhugile mujale eraldi (nt mõnda teise akna osasse või lausa eraldi programmi).

Mis on rasvakriit?

"Rasvakriidi" (*Grease Pencil*) nimi on kumardus esimestele arvutianimaatoritele, kes kasutasid vahast kriite/pliiatseid, et oma kinoskoopkuvarite ekraanile kõveraid ja teisi planeerimismärkeid joonistada.

Lisaks animaatoritele, kes saavad rasvakriidi abil planeerida poose ja liikumistrajekteore, võib see töörist olla kasulik ka paljudes teistes olukordades. Muuhulgas:

- mudelite paigutuse ja topoloogia planeerimiseks
- režissööri tööriistana võttenurkade arvustamiseks
- "tahvi" ja ülesannete tööriistana õpetajatele.

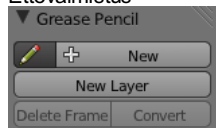


Rasvakriidiga töötamine.

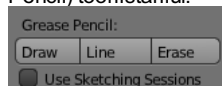
Järgmised leheküljed seletavad, kuidas seda tööriista kasutada:

- [Visandite joonistamine.](#)
- [Kihid ja animeerimine.](#)
- [Visandite muutmine geomeetriaks.](#)

Ettevalmistus



Rasvakriidi (Grease Pencil) tööriistariiul.



Rasvakriidi (Grease Pencil) sätete paneel.

- Esimene samm rasvakriidi kasutamisel on rasvakriidi esitamise sisse lülitamine. Otsi üles rasvakriidi (Grease Pencil) märges tööriistariiulil (vajuta T, kui riul pole avatud) ja Literaalsätete paneelis (vajuta N, kui paneel pole avatud).
- Seejärel vajuta nupule New Layer, mis lisab joonistamiseks uue kihi. See samm ei ole vajalik, kui alustad uue joonistusega (sest uus kiht tehakse sinu jaoks automaatselt), kuid siit saad ka enne joonistama asumist muuta joone laiust, värvi ja läbipaistvust. Kui tahad joonistada uuele kihile ja mõned kihid on juba olemas, on sellele nupule vajutamine vajalik.

Rasvapliiatsit saab kasutada järgnevates redaktori (akna) tüüpides: 3D-vaade (3D View) ja UV/Pildi redaktor (UV/Image Editor).

Vabajoone tõmbamine

Mode: Kõik režiimid

Panel: Object Tools » Grease Pencil » **Draw**

Hotkey: D LMB

Kirjeldus

Tõmba uus vabakäeline joon (mis koosneb paljudest omavahel seotud väikestest joonekestest). Joon lõppeb, kui lased hiire klahvi lahti.

Vihjed

Hoides all nuppu D, vajuta punkti tegemiseks korra klahvi LMB .

Sirgjoone tõmbamine

Mode: Kõik režiimid

Panel: Object Tools » Grease Pencil » **Line**

Hotkey: CtrlD LMB

Kirjeldus

Tõmba uus joon kummipaelarežiimis. Joon lõppeb, kui lased hiire klahvi lahti.

Kustukumm

Mode: Kõik režiimid

Panel: Object Tools » Grease Pencil » **Erase**

Hotkey: D RMB

Kirjeldus

Kustutab joonte osad, mis jäävad kustutaja "pintsl" raadiusse. Kustutamine kestab hiire klahvi lahti laskmiseni.

Sätted

Kustukummi "pintsl" raadiust saad muuta menüüst User Preferences » Editing » Grease Pencil » Eraser Radius.

Vihjed

Kui vajutad nupule **Erase**, siis kustutatakse jooni nii klahvi RMB kui ka klahvi LMB all hoides.


Visandiseansid

Mode: Kõik režiimid

Panel: Object Tools » Grease Pencil » Use Sketching Sessions

Kirjeldus

Visandiseansiga (*Sketching Session*) saad rasvakriidiga kiiresti skitseerida, kui soovid teha mitut tõmmet järjest. Kui see valik on

sisse lülitatud, algab pärast rasvakriidiga joone tõmbamist visandiseanss. Iga kord, kui vajutad klahvi **LMB** , tõmmatakse uus sama tüüpi joon. Seansist väljumiseks vajuta Esc või ↵ Enter.

Vihje

Kustutamiseks loetakse nii klahvi **LMB**  kui ka klahvi **RMB** .

Jagatud rasvakriidi sätted

Joonistamise sätted

Menüüs Transform Panel » Grease Pencil » Drawing Settings on kuni neli joonistamise seadistust. Neid sätteid saab iga rasvakriidi kihi siseselt segamini kasutada.

- Vaade **View** - Uued jooned lukustatakse valitud vaatesse. Need jooned *ei ole* kolmemõõtmelises ruumis ja neid võib pigem pidada ekraani peale joonistatuteks.
- Kursor **Cursor** (ainult 3D-vaates) - Uued jooned joonistatakse kolmemõõtmelisse ruumi ning nende asukoht sõltub nende tegemise aegsest 3D-kursori asukohast ja vaate pöördenurgast (st nad asuvad vaatesuunaga ristuva tasapinnal ja läbivad 3D-kursorit).
- Pind **Surface** (ainult 3D-vaates) - Need jooned joonistatakse kolmemõõtmelisse ruumi ja nende asukoht sõltub nende "alla" jäävast esimesest pinnast antud vaates (st projitseeritakse sellele pinnale).
- Joon **Stroke** (ainult 3D-vaates) - Uued jooned joonistatakse kolmemõõtmelisse ruumi ning nende asukoht sõltub juba varem kolmemõõtmelisse ruumi joonistatud joonte asukohtadest, mis jäävad tõmmatava joone *alla* (st nad projitseeritakse juba olemasolevatele joontele). Pea meeles, et valikuga View tõmmatud jooned ei ole kolmemõõtmelises ruumis ning seega nendega projitseerimise puhul ei arvestata.

Kui kasutad sätteid Surface või Stroke ja ükski uue joone osa ei *maandu* mõnel pinnal või olemasoleval joonel, siis uut joont ei salvestata. Kui "maandub" kas või mõni uue joone osa, siis salvestatakse terve joon.

Pannes linnukese kasti Transform Panel » Grease Pencil » Drawing Settings » Only Endpoints, rakendatakse joonistamise seade ainult joone algus- ja lõpppunktile ning kõiki vahepealseid punkte muudetakse sedasi, et nad langeksid neid kahte punkti läbivale tasapinnale.

Piirangud

Kursori valik (Cursor) on kasutatav ka UV/Pildi redaktoris (UV/Image Editor), kuid töötab samamoodi nagu vaate valik (View) — milleks joone tüüp tõmbamise lõppedes ka muutub.

Joonistamise tundlikkus

Hiire/krihvli (stiluse) vaikimisi tundlikkus on selline, et joon joonistamise ajal eriti ei väriseks, kuid samas oleks ikkagi võimalik teha peeneid detaile. Mõnikord aga ei pruugi need seaded sobida ning siis leiad need valikud menüüst User Preferences window » Editing » Grease Pencil.

- Manhattan Distance: minimaalne pikslite arv, mille hiir peab horisontaalselt või vertikaalselt läbima, enne kui seda joonistuses arvestatakse. Selle numbri vähendamine peaks parandama kõveraid jooni.
- Euclidean Distance: minimaalne hiire teekond, enne kui liigutust joonistuses arvestatakse. Seda võib ette kujutada kui hiire kursori ja viimati salvestatud asukoha vahel olevat kujuteldavat niiti.
- Eraser Radius: kustukummi raadius. See tähistab lihtsalt kustukummi "pintsl" raadiust ning selle muutmine mõjutab seda, millise tööenäosusega jäävad jooned kustukummi raadiuse sisse ja saavad kustutatud.
- Smooth Stroke: see valik lülitab sisse järeltöötluse sammu, mille jooksul muudetakse jooni siledamaks, et võbelusi vähendada. See ei mõjuta sirgete joonte tõmbamist. Vaikimisi on see sisse lülitatud. Siinkohal tuleb märkida, et see valik võib sageli põhjustada joonistuste "kokkutõmbumist" ja seega võib selle välja lülitada, kui tulemus ei meeldi.

Digilaua kasutajatele

- Joone paksust igas punktis mõjutab joonistamisel parajasti rakendatud surve.
- Krihvli "kustukummi" otsa saab kasutada joonte kustutamiseks.

Tasapindade joonistamine

Lisainformatsioon versiooni 2.5x kohta

Märkmena neile, kes hakkavad seda lehte päriselt muutma, jätan ma järgneva video

[\[video link\]](#)

Kihid

Rasvakriidi visandid on organiseeritud kihtideks, mis sarnanevad GIMP-i ja Photoshopi omadega. Need kihid ei ole seotud ühegi teise Blenderi kihtide süsteemiga ning seetõttu pole nende hulk ka maksimaalse kihtide arvuga piiratud. Nagu eelpool mainitud programmide kihte, saab ka rasvakriidi omi nimetada, lukustada, peita ja kustutada.

Kihtide peamiseks eesmärgiks on koguda kokku mingit sorti ühiseid omadusi sisaldavad visandid (nt "tegelaste asukohad", "režissööri kommentaarid asukohtadele" või "juhendid"). Sel põhjusel mõjutavad ka kihi värvile, läbipaistvusele ja joone jämedusele tehtavad muudatused kõiki selle kihi jooni (ja mitte ainult neid jooni, mis on tõmmatud pärast mõne sätte muutmist).

Vaikimisi toimub enamus tegevust *aktiivse*l kihil. Aktiivset kihti saab teistest eristada teistsuguse paneelivärviga (vaikimisi on selleks hele oranžikaspruun). Kihil klõpsamine või mõne selle omaduse muutmine teeb selle aktiivseks.

Aktiivset kihti saab kindlaks teha ka olekunäidiku kaudu (iga rasvakriidiga vaate ülemises parempoolses nurgas).

Visandite animeerimine

Rasvapliiatsit saab kasutada lihtsate testjoonistuste tegemiseks (st kahemõõtmeline animatsioon, mida koolis vihikunurkadesse kritseldasid). Visandid salvestatakse kaadrisse, kus nad tehti, täiesti eraldiseisva joonistusena (ainult sellel kihil, kus nad ise on). Iga joonistus on nähtav seni, kuni programm kohtab selle kihi järgmist joonistust. Erandiks on kihi esimene joonistus, mis on nähtav ka enne kaadrit, millesse ta joonistati.

Seeläbi on lihtne teha animeeritud visandite seeria testi:

1. Mine esimesse vajalikku kaadrisse. Joonista.
2. Mine järgmisse kaadrisse. Joonista veel.
3. Jätka sama protsessi ja joonistamist seni, kuni oled tulemusega rahul. *Voilà!* Animeeritud visandid.

Sibulakoore meetod

Sibulakoore meetod (*onion skinning* või *ghosting*) on animaatoritele väga kasulik, sest selle aktiveerimisel näitab Blender ähmaselt ka naaberkaadrite sisu. See aitab animaatoril paremini liigutusi hinnata, sest võimaldab erinevates kaadrites olevaid etappe võrrelda.

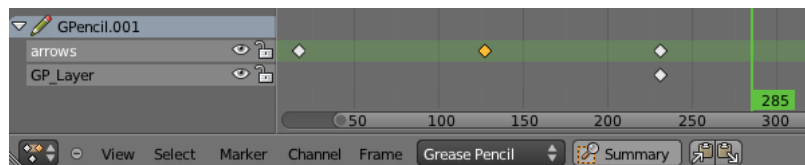
Märkmed kasutamise kohta:

- Sibulakoore meetodit saab igal kihil eraldi sisse lülitada, vajutades rasvakriidi sätete paneelis olevat nuppu Onion Skin.
- Kaadrite lahter Frames nupu Onion Skin all näitab, mitu kaadrit joonistatakse. Kui Frames on **0**, näidatakse ainult hetkekaadri eelnevat ja järgnevat kaadrit. Muu väärtuse puhul määrab see lahter praegusest kaadrist mõlemale poole jäävate kaadrite arvu, mida tuleb näidata.

Visandite ajastuse muutmine

Rasvakriidi visandite plokki saab laadida töölehele (DopeSheet), et seal joonistuste ajastust muuta. See on eriti kasulik animaatoritele, kes tegelevad tegelaste liikumisega, mille puhul on ajastuse muutmine üks nende töö tähtsamaid osasid.

1. Muuda töölehe aknas (DopeSheet) menüüde kõrval asuv režiimi valik rasvapliiatsiks (Grease Pencil) (vaikimisi peaks see olema seatud töölehe (DopeSheet) enda peale).
2. Nüüd peaks tööleht DopeSheet näitama mõningaid "kanaleid" koos nendel olevate "võtmekaadritega". Need "kanalid" on kihid ja "võtmekaadrid" on kaadrid, kus asub mõni visanditest. Nendega saab manipuleerida nagu kõigi ülejäänud töölehel DopeSheet olevate andmetega.



Näidatakse kõiki parajasti valitud stseenis olevaid rasvakriidi plokkide. Ala/rasvakriidi andmeplokid on joonistatud roheliste kanalitega ning nimetatud vaadetest pärit info järgi. Nende nimes sisaldub ka ala (st akna) indeks (kuigi seda kuskil mujal praegu ei näe).

Visandite kopeerimine

Tegevuse halduris (Action Editor) on võimalik visandeid ühelt või mitmelt kihilt teistele kopeerida, kasutades selleks päises olevaid kopeerimise/kleepimise nuppe *Copy/Paste*. See töötab nagu võtmekaadrite kopeerimine/kleepimine tegevuste halduris (Action Editor).

Visandeid saab nende tööriistade abil kopeerida ka ühelt ekraanilt (või vaatest) teise. Oluline on meeles pidada, et võtmekaadrid kleebitakse ainult valitud kihtidele ja seetõttu tuleb kleepimiseks need kihid kõigepealt luua.

Visandite teisendamine teisele kujule

3D-vaates saab aktiivsel kihil olevad visandid muuta geomeetriaks, mille tulemus sõltub parajasti valitud vaate sätetest. Visandite geomeetrilisteks objektideks teisendamine kasutab joone tõmbamisel saadud punkte ning teisendab need 3D-ruumi (vastavalt aktiivse vaate sätetele). Praegusel hetkel kasutatakse kõiki punkte ning seetõttu võib tavakasutuses tekkida vajadus loodud objekte lihtsustada või osadeks jagada.

Visandeid saab praegusel hetkel muuta kaheks erinevaks objektitüübiks, vastavalt rasvakriidi omaduste paneeli nupu Convert... vajutamise järel lahti hüppavast menüüst Grease Pencil Convert:

- **Rada** (*Path*)
- **Bézier'i kõver** (*Bezier Curve*)

Iga joon muudetakse eraldi kõveraks ühes kõverobjektis, mis saab nime aktiivse kihi järgi. Sangad on automaatselt "vabad" (st musta tüüpi) ning need asuvad samades kohtades nagu juhtpunktidki. Kõvera raskus/raadius igas juhtpunktis võrdub iga salvestatud joone punkti paksusega. Kuid selleks, et seda näha, pead muutmisrežimis Edit avama teisendussätete paneeli Transform Properties ja vaatama kaalu välja Weight.

Võreks muutmine

Kui soovid muuta oma visandit võreks, vali esmalt Bézier'desse muutmine (Active Layer to Bezier) ning muuda seejärel loodud kõver võreks...

Blenderiga modelleerimine

Nagu oli näha [sissejuhatavast](#) peatükist, on 3D-stseeni loomiseks vaja vähemalt kolme tüüpi asju: mudeleid, materjale ja valgusteid. Selles osas kaevume me sügavamalt esimesse teemasse: modelleerimisse. Modelleerimine on nii kunst kui ka teadus, mille abil luuakse pärismaailmas eksisteerivate objektide sarnased kujundid või mille abil kujutatakse abstraktseid, vaid ettekujutuses eksisteerivaid objekte.

Objekte on igasuguse kuju ja suurusega. Blenderis on palju erinevaid tööriistu, mis aitavad mudeli kiirelt ja efektiivselt valmis teha:

[Objektid \(*Objects*\)](#)

Objekti kui terviku töötlemine

[Võred \(*Meshes*\)](#)

Objekti kuju määrava võrega töötamine

[Kõverad \(*Curves*\)](#)

Töö kõverobjektidega ja kõveratel põhinevate radadega

[Pinnad \(*Surfaces*\)](#)

NURBS-objekti loomine

[Tekst \(*Text*\)](#)

3D-s tekstiobjektidega töötamine

[Metaobjektid \(*Meta Objects*\)](#)

Kerad ja tilgad

[Duplikaadid \(*Duplications*\)](#)

Objektidest koopia tegemine ja objektide omavahelised asendused

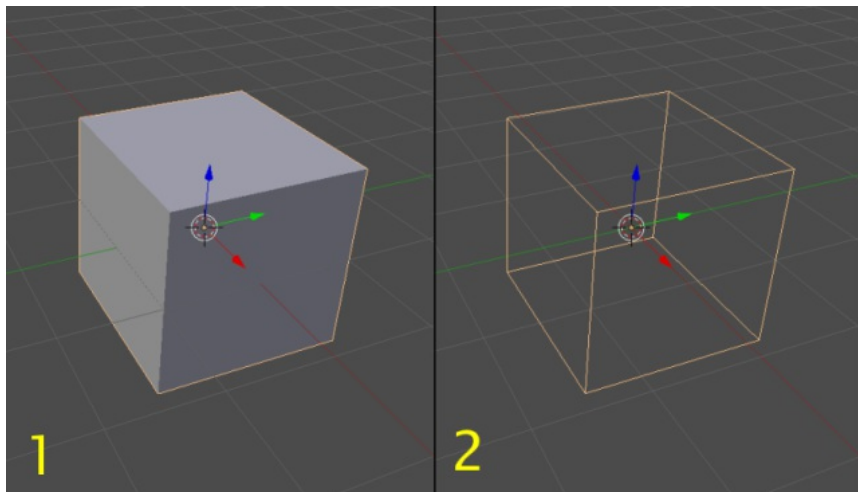
[Pythoni koodijupid](#)

Kuna Blenderi kasutajaliides ja kõik võimalused on kättesaadavad ja laiendatavad Pythoni (üks programmeerimiskeel) abil, saab Pythoni skriptidega Blenderi funktsionaalsust veelgi suurendada.

Paljud inimesed harrastavad "kuubikust modelleerimist", mis algab kuubikuga, jätkub külgede eendamisega ja tippude liigutamisega, kuni lõpuks on tegemist keeruka võrega. Lamedate objektide, näiteks seinte ja laudade puhul, võib kasutada kõveratega modelleerimist. Kuju antakse siis Bezier' või NURBS-kõveratega ja eendatakse seejärel soovitud paksuseni. Mõlemad variandid on Blenderis täiesti olemas.

Objektirežiim (*Object Mode*)

Steeni geometria koosneb ühest või enamast objektist: näiteks valgustitest, kõveratest, pindadest, kaameratest, võredest ja põhivormidest, mis on kirjeldatud teemas "[võreprimitiividdest](#)".



Valitud objekt.

Objektide tüübid

Võred (Meshes)	Võreobjektid koosnevad polügonaalsetest külgedest, servadest ja/või tippudest ning neid saab paljudel viisidel Blenderi tööriistadega töödelda.
Kõverad (Curves)	Kõverad on matemaatilised objektid ja nende kuju saab lisaks tippudele muuta ka sangadega ehk tippude kontrollpunktidega.
Pinnad (Surfaces)	Pinnad ehk pindobjektid on nelja küljega latakad, mida saab samuti muuta kontrollpunktidega. Neid on mugav kasutada orgaaniliste ja ümarate, aga lihtsate kujundite puhul.
Metaobjektid (Meta Objects)	Metaobjektid on objektid, mille kuju ja maht on määratud matemaatiliste funktsioonidega. Metaobjektidega saab tekitada ebamäärase kujuga ümaraid vorme. Kahe metaobjekti olemasolu korral käituvad need pisut vedeliku moodi ja kleepuvad üksteise külge pindpinevusjõu sarnaselt.
Tekst (Text)	Tekstiobjektide abil saab luua tekste.
Skeletid (Armatures)	Skelettide abil saab 3D-mudelid muuta poseeritavaks ja animeeritavaks.
Tühiobjektid (Empty)	Tühiobjektid on abivahendid, mida saab kasutada teiste objektide teisendamiseks ja allutamiseks. Neid ei renderdata. Neid saab kasutada näiteks objekti asukoha või liikumise muutmiseks.
Kaamerad (Cameras)	See on virtuaalne kaamera, mille abil saab ära määrata, mida renderdataval pildil näidatakse.
Lambid (Lamps)	Nende abil saab stseeni valgustada.
Jõuväljad (Force Fields)	Jõuvälju kasutatakse füüsika simulatsioonide puhul. Nende abil saab väliseid jõudusid simuleerida (liikumist tekitada) ja neid kuvatakse väikeste kontrollobjektidena.



Igat objekti saab objektirežiimis nihutada, keerata ja suurendada. Siiski mitte kõik teisendused ei anna kõigi objektide puhul tulemust. Näiteks jõuvälja (*force field*) suurendamine selle mõju ei suurenda.



Muudetava objekti muutmiseks tuleb siseneda muutmisrežiimi (Edit mode).

Objekti lisamisel ollakse vaikimisi objektirežiimis (Object Mode). Varasemates Blenderi versioonides oli nii, et pärast objekti lisamist satuti koheselt muutmisrežiimi. See kehtis võre-, kõver- ja pindobjektide puhul.

Objektirežiimi (Object Mode) ja muutmisrežiimi (Edit Mode) vahel lülitumiseks kasuta klaviatuuri klahvi \leftrightarrow Tab.

Kui objektil on olemas võre, muutub see nüüd oranžiks. Nii nagu näidatud pildil *Valitud objektid*.

Pilt "Valitud objektid" näitab vaikekuubikud tahkes (*solid*) vaates ja traatvõre (*wireframe*) vaates. Traatvõre ja tahke vaate vahel lülitumiseks kasuta klahvi Z.

Objekti keskpunkt ehk kese ehk asukohapunkt (*Object Centers*)

Igal objektil on olemas keskpunkt või siis teisisõnu asukoha punkt. Selle punkti järgi saab öelda, mis kohas objekt täpselt koordinaatteljestikul asub. Kui objekt on valitud, siis on näha väikest ringikest, mis näitab objekti asukohta. Objekti kaldenurga muutmisel ja suurendamisel on see punkt väga tähtis. Vaata täpsemalt ka [objekti keskpunkti](#) alateemat.

Objekti keskmee muutmine

Objekti keset saab muuta läbi objekti menüü Object teisendamise alammenüü Transform:

- 3D Cursor Location (3D-kursori asukoht)

See liigutab objekti keskme 3D-kursori asukohta. 3D-kursori täpsema kasutamise kohta vaata teemat [3D-vaate kasutamine](#).

- *Median Point of Geometry* (geomeetria mediaanpunkt)

Liigutab keskpunkti objekti piirdkasti geomeetrilisse keskmesse.

Objektide kustutamine

Mode: Muutmisrežiim Edit või objektirežiim Object

Hotkey: X või Del

Menu: Object → Delete

Kustutab või eemaldab valitud objektid.

Objektide ühendamine (*Join Objects*)

Mode: Objektirežiim Object

Hotkey: CtrlJ

Menu: Object → Join Objects

Ühendab kõik valitud ja sama tüüpi objektid üheks objektiks kokku. Uus "keskpunkt" on määratud viimasena valikusse võetud objekti keskpunktiga. Ühendamine on üsnagi sarnane sellise tegevuse puhul, kui lisatakse uus objekt, olles samal ajal muutmisrežimis (Edit mode). Mitteaktiivsed objektid kustutatakse ja aktiivne objekt jääb alles. See toimib ainult muudetavate objektide puhul, nagu näiteks võred ja kõverad.

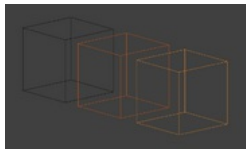
Sissejuhatus

Peaaegu igas programmis tähendab valimine määramist, millised elemendid on meie tegevuse sihtmärgiks. Seetõttu on seda parem, mida sobilikum on valimise tööriist soovitud tegevuse jaoks. Blenderis on väga palju tööriistu ja funktsioone ning seega on palju ka valimise meetodeid.

Järgnevalt anname lühiülevaate mõistetest ja valimise tööriistadest, mida saab kasutada objektirežiimis Object.

Valikud ja aktiivne objekt

Blender teeb vahet kahel valimise seisundil:



Valimata objekt mustana, valitud objekt oranžina ja aktiivne objekt kollasena

- Objektirežiimis Object nimetatakse objekti, mis valiti viimasena või mille valik tühistati viimasena, "aktiivseks objektiks" ja see on kollase kontuuriga (teiste kontuur on oranž). Alati on ainult üks aktiivne objekt (isegi siis, kui midagi pole valitud!).

Mitmed Blenderi tegevused, näiteks linkimine, kasutavad alusena aktiivset objekti. Kui sul juba on valik, kuid tahad teha teist objekti aktiivseks, vali see lihtsalt kombinatsiooniga **⇧ Shift RMB** uuesti.

- Kõik teised valitud objektid ongi täpselt seda - valitud. Valitud objektide hulk ei ole piiratud.

Punktvalik

Kõige lihtsam objekti *valimise* viis on sellele klahviga **RMB** vajutamine.

Et objekte *valikusse lisada*, kasuta kombinatsiooni **⇧ Shift RMB** teiste objektide peal.

Kui *objektid vaates kattuvad*, saad kasutada kombinatsiooni **Alt RMB**, et näha võimalike valikute nimekirja.

Kui soovid sedaviisi *valikule lisada*, on kiirvaliku kombinatsiooniks **⇧ ShiftAlt RMB**.

Et *aktiveerida objekti*, mis on juba valitud, vajuta selle peal kombinatsiooniga **⇧ Shift RMB**.

Aktiivse objekti *valiku tühistamiseks* vajuta sellele üks kord kombinatsiooniga **⇧ Shift RMB** - ja kui objekt pole aktiivne, siis tuleb seega vajutada kaks korda (aktiveerimiseks ja valiku tühistamiseks).

Nelinurga ehk piirdkastiga valimine

Mode: Objektirežiim Object

Hotkey: B

Menu: Select → Border Select

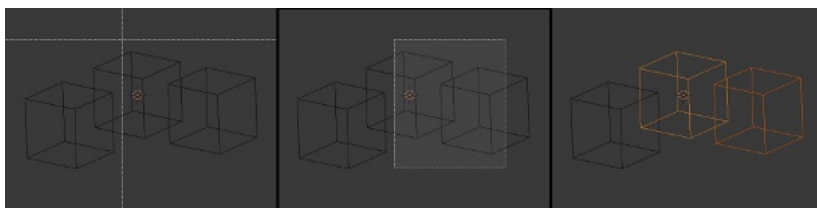
Kirjeldus

Piirdkastiga Border Select valides joonistad sa klahvi **LMB** all hoides nelinurga. Iga objekt, mis jääb kas või osaliselt selle nelinurga sisse, muutub valituks.

Objektide valiku tühistamiseks vajuta klahvi **MMB**.

Valiku tühistamiseks kasuta klahvi **RMB**.

Näide



Piirdkastiga valimine kolme sammuna

Pildil (*Piirdkastiga valimine...*), on esimeses osas piirdkast Border Select aktiveeritud ning sellele viitab punktiirina näidatav hiirekursor. Teises osas tekitatakse *valimise ala*, joonistades klahviga **LMB** nelinurk. See katab vaid kahte kuupi. Viimasel pildil on valik pärast klahvi **LMB** lahti laskmist sooritatud.


Pane viimasest osas tähele kõige vasakpoolsema valitud kuubi eredat värvi. See tähendab, et tegemist on "aktiivse objektiga" - enne piirdkastiga Border Select valimist viimati valitud objektiga.

Vihje

Piirdkastiga valimine lisab eelmisele valikule, seega kui soovid valida ainult nelinurga sisse jäävaid objekte, pead enne nupuga A kõik varasemad valikud tühistama.

Lassoga valimine

Mode: Objektirežiim Object

Hotkey: Ctrl LMB 

Menu: menüü sissekannne puudub

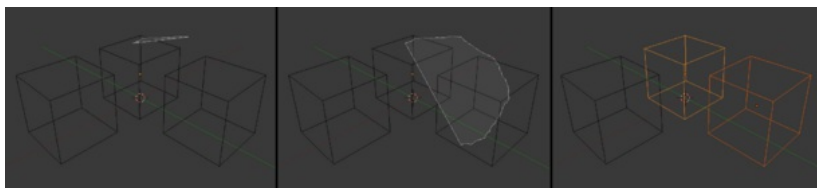
Kirjeldus

Lassoga valimise puhul tõmmatakse objektirežiimis Object ümber valitavate objektide pöörlemistelgede (ehk kinnituspunkti) punktiirjoon. Valitakse vaid need objektid, mille pöörlemistelg jääb märgitud ala sisse.

Kasutamine

Hoides all klahvi Ctrl, tuleb lihtsalt joonistada hiireklahviga LMB  joon ümber soovitatavate objektide pöörlemistelgede.

Lasso valik lisandub varasemale valikule. Valiku tühistamiseks kasuta kombinatsiooni Ctrl⇧ Shift LMB .



Lassoga valimise näide

Menüüga valimine

Ülalpool kirjeldatud valikumeetodid on kõige tavapärasemad. 3D-vaate valikumenüüs Select on lisaks veel palju erinevaid võimalusi.

Igaüks neist on mõeldud mingite kindlate tegevuste jaoks.

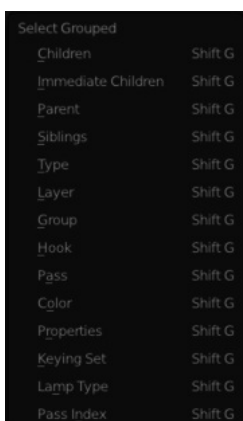
Vali grupeeritud (*Grouped*)

Mode: Objektirežiim Object

Hotkey: ⇧ ShiftG

Menu: Select → Grouped

Kirjeldus



Grupeeritud valiku menüü Grouped

On kaks moodust objekte omavahelisse sõltuvusse seada. Esimene on hierarhilised suhted ja teine grupeerimine. Neid suhteid saame me ära kasutada, valides soovitud hierarhiate või gruppide liikmeid.

Valikud

Menüü Select → Grouped kasutab objektirežiimis Object aktiivset objekti kõigi teiste valimise alusena.

Võimalikud valikud on:

Children (alamad)

Valib kõik aktiivse objekti alamad, kaasa arvatud alamate alamad jne.

Immediate Children (otsesed alamad)

Valib kõik aktiivse objekti otsesed alamad.

Parent (ülem)

Valib objekti ülema, kui see eksisteerib.

Siblings (sama tasand)

Valib aktiivse objektiga sama ülemat jagavad objektid. Sellega saab samuti valida kõik juure tasandil olevad objektid (objektid, millel pole ülemaid).

Type (tüüp)

Valib kõik objektid, mille tüüp on sama nagu aktiivsel objektil.

Layer (kiht)

Objektid, millel on vähemalt üks jagatud kiht.

Group (grupp)

Valitakse objektid, mis kuuluvad gruppidesse (vaikekasutajaliidese kuvamise puhul on nad rohelised), milles on ka aktiivne objekt.

Object Hooks (objekti haagid)

Valitakse kõik aktiivsele objektile kuuluvad haagid (*hooks*).

Pass (käik)

Valitakse objektid, mis on määratud samasse renderduskäiku (*render pass*). Renderduskäike saab määrata paneelis Properties → Object → Relations ning neid saab kasutada komposiitoris Node Compositor (Add → Convertor → ID Mask.)

Color (värvus)

Vali objektid, mille värvus Object Color on sama. Objekti värve saab määrata paneelis Properties → Object → Display → Object Color.)

Properties (omadused)

Vali objektid, mille mängumootori omadused (Game Engine Properties) on samad.

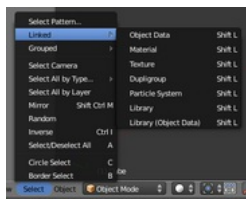
Vali lingitud (*Linked*)

Mode: Objektirežiim Object

Hotkey: ⇧ ShiftL

Menu: Select → Linked

Kirjeldus



Lingitud objektide valimise menüü Linked

Valitakse kõik objektid, mis jagavad aktiivse objektiga ühist andmeblokki.

Valikud

Menüü Select → Linked kasutab objektirežimis Object aktiivset objekti kõigi teiste valimise alusena.

Võimalikud valikud on:

Object Data (objekti andmed)

Valib kõik objektid, mis on lingitud samade objekti andmetega, st andmeblokk, mis määrab objekti tüübi (võre, kõver jne), ja ehituse (objekti määravad elemendid nagu tipud, kontrolltipud ning see, kus nad ruumis asuvad).

Material (materjal)

Valib kõik objektid, mis lingivad samale materjali andmeblokkile.

Texture (tekstuur)

Valib kõik objektid, mis lingivad samale tekstuuri andmeblokkile.

Dupligroup (dupligrupp)

Valib kõik objektid, mis kasutavad duplitseerimiseks sama gruppi.

Particle System (osakeste süsteem)

Valib kõik objektid, mis kasutavad sama osakeste süsteemi

Library (teek)

Valib kõik objektid, mis on samas [teegis \(Library\)](#)

Library (Object Data) (objekti andmete teek)

Lisavõimalused

Tööriistariiulis Tool Shelf on olemas järgnevad võimalused:

Extend (laienda)

Võimaldab lisada objekte praegusele valikule, selle asemel et valik nendega asendada.

Layer (kiht)

Kiht, millel objektid asuvad.



Objektide valimine

Selle asemel et kasutada kihi kaupa valimise võimalust Select All by Layer, võib olla kasulik muuta vajalikud kihid nähtavaks ning valida kõik nähtavad objektid (A). Selle meetodi puhul saad ka objektide valikut tühistada.

Teised menüü võimalused

Menüü esimesel tasemel on veel järgnevad võimalused:

Random (juhuslik)

Valib juhuslikult parajasti aktiivsete kihtide valimata objektide hulgast objektid vastavalt määratud tõenäosusele. Selle käsu valimisel lisandub tööriistariiulisse Tool Shelf numbriline valikukast.

Oluline on märkida, et see protsent tähistab valimata objekti valimise tõenäosust, mitte protsenti objektide koguarvust.

Inverse (tagurpidine valik, Ctrl)

Valib kõik objektid, mis ei olnud valitud, ning tühistab varem valitud objektide valiku.

Vali kõik/tühista kõigi valik (Select/Deselect All, A)

Kui miski oli valitud, siis kõigepealt see valik tühistatakse. Muudel juhtudel see kas valib kõik nähtavad objektid või tühistab valiku.

Border Select (piirdkastiga valik, B)

Nagu seda kirjeldati [piirdkastiga valimise osas](#).

Objektide teisendamine

Objekte saab erinevatel viisidel teisendada. Seda on korralikult kirjeldatud järgmises peatükis [Manipulatsioonid 3D-ruumis](#). Allpool on äratoodud erinevad võimalikud teisendused objektirežiimis Object:

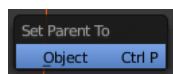
- Liigutamine
- Pööramine
- Mõõtkava teisendamine
- Peegeldamine

Objektide grupeerimine ja allutamine

Steenis võib olla palju objekte: tüüpiline lavastseen koosneb mööblist, rekvisiitidest, valgustitest ja taustadest. Blender aitab asju paremini organiseerida läbi objektide grupeerimise.

Näiteks sellise keerulise objekti nagu kella puhul võib jupid eraldi objektidena modelleerida. Kõiki neid objekte saab üksteise külge ühendada. Sellisel juhul võib määrata, et üks objekt on ülemobjekt ja teised alluvad sellele. Ülemobjekti keeramine keerab sellega koos ka alamobjekte.

Objektide allutamine



Allutamise)
hüplikmenüü Set
Parent To

Objekti allutamiseks vali vähemalt kaks objekti (esimesena vali alluv ja seejärel ülem) ja vajuta CtrlP. Ilmub dialoogiaken, kus küsitakse, kas soovid allutada (Make Parent), ja tahetakse tegevusele kinnitust saada. Vajutades Make Parent, kinnitatakse tegevus ja toimub alluvusseose loomine. Viimasena valitud ehk aktiivseks tehtud objektist (kollase äärejoonega tähistatud) saab ülemobjekt. Mitme objekti valimisel saab ülemobjektiks viimasena valitu. Kõik teised on sellega võrdses alluvussuhtes, n-ö. hierarhias samal astmel.

Alluvate toomiseks ülema sisse vajuta ⇧ ShiftCtrlP. Tulemuseks on alternatiivne alluvussuhe, kus kõik alluvad on nüüd seotud otseselt ülemobjekti koordinaatsüsteemiga. See on hea näiteks projekteerimise puhul.

Ülemobjekti keeramine ja liigutamine keerab ja liigutab tavaliselt ka alluvaid objekte. Samas alluvobjekti liigutamine/keeramine ülemobjekti ei liiguta/keera. Ühesõnaga, mõju on ühesuunaline (ülem → alluv/alluvad) ja teisesuunaline mõju puudub (alluv/alluvad → ülem).

Allutamine tippudele

Objekti võib allutada ka tippude grupile (*vertex group*) või siis lihtsalt minimaalselt kolmele tipule; alluvobjektid näevad siis ülemobjekti deformeerimisel välja nagu sääsk tuksleval arteril. Vali objektirežiimis tulevane alluv ja seejärel tulevane ülemobjekt. Vajuta ⇆ Tab klahvi, et minna muutmisrežimi ja vali ära üksik tipp või 3 tippu (need ei pea omavahel päris külge moodustama); seejärel vajuta CtrlP ning kinnita käsk.

Kui valitud oli üks tipp, siis nüüd kuvatakse sellest tipust alluvobjekti suunas katkendjoon. Kui valitud oli kolm tippu, kuvatakse katkendlik joon alluvobjekti nende tippude keskkohast. Tulemuseks on nüüd see, et kui ülemobjekt deformeerub ja muutub valitud tippude asukoht, siis liigub ka alluvobjekt.

Märkus

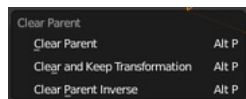
See on nagu tagurpidi toimiv [haak \(hook\)](#).

Valikud

Alluvobjekti liigutamine

Alluvobjekti saab viia ülemobjektiga samasse asukohta, kui kustutada tema "keskpunkti" (*origin*) asukoha info. Alluvussuhe sellest ei muutu. Vali alluvobjekt ja vajuta AltO. Pärast käsu dialoogiaknas kinnitamisest liigub alluvobjekt ülemobjekti asukohta. Veendumaks, et alluvussuhe ei muutunud, kontrolli seda struktuuriredaktori Outliner vaates.

Alluvussuhte muutmine ja tühistamine



Alluvuse tühistamise
hüplikmenüü Clear Parent

Alluvussuhte saab tühistada klahvikombinatsiooniga AltP.

Ilmunud menüüs on järgmised valikud:

Clear Parent (tühistab alluvuse)

Kui valitud oli ülemobjekt, ei juhtu selle valiku tulemusel midagi. Kui valitud oli alluvobjekt, siis selle tulemusel alluvusseos kaotatakse ja objekt saab tagasi oma "algse" asukoha, pöörde ja suuruse.

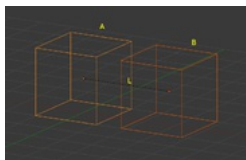
Clear and Keep Transformation (tühistab alluvuse ja hoiab teisendusi)

Vabastab alluva ülemusest, aga jätab alles tollelt saadud asukoha, pöörde ja suuruse.

Clear Parent Inverse (alluvussuhte pööramine)

Jätab alluvussuhte alles, aga paigutab alluva ülema suhtes nii, nagu nad asetsesid enne alluvussuhte loomist. Nii saab ülemobjekti eelneva mõju ära nullida. Näiteks kui ülemobjekti nihutati 10 ühikut X-teljel ja alluvobjekt liikus sama palju kaasa, siis Clear Parent Inverse nihutab alluvobjekti algseks asukohta -10 ühikut X-teljel tagasi. Alluvussuhte pööramine kasutab ainult viimast muutmist; kui ülemobjekti nihutati 2 korda, 10 ühikut mõlemal korral, siis "inverse" nihutab alluvat tagasi vaid 10, mitte 20 ühiku võrra.

Allutamise näide

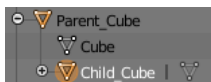


Allutamise näide

Aktiivne objekt, kollasega tähistatult (kuubik A), saab kõigi grupis olevate objektide ülemobjektiks (oranž kuubik B). Kõigi alluvobjektide "keskpunktid" on nüüd seostatud ülemobjekti "keskpunktiga" ja on näha neid ühendavad katkendjooned; vaata pilti ("Allutamise näidis"). Ülemobjekt on kuubik "A" ja alamobjekt samuti kuubik "B". Seos on tähistatud "L"-tähega.

Nüüd on nii, et ülemobjekti nihutamine, keeramine ja suurendamine teeb sama ka alluvobjektidega. Allutamine on mitmete rakenduste juures väga oluline. Seda kasutatakse väga palju ka keerulisemate animatsioonide puhul.

Soovitused



Struktuuriredaktori (Outliner) vaade

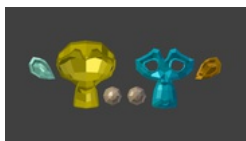
Alluvussuhet võib vaadata ka struktuuriredaktori (Outliner) kaudu [struktuurivaate](#) aknas. Pilt (*Struktuuriredaktori (Outliner) vaade*) on näiteks selle kohta, millisena näeks *Allutamise näide* välja struktuuriredaktori vaates. Kuubiku "A" objekti nimi on "Cube_Parent" ja kuubiku "B" nimi "Cube_Child".

Objektide eraldamine

Mingil hetkel jõuad sa selleni, et on vaja objektist mingi osa eraldi objektiks eraldada. See liigutus on lihtne.

Eraldamiseks peavad olema valitud soovitud tipud või küljed. Pärast valimistoimingu tegemist on eraldamiseks mitu erinevat varianti. Muutmisrežiimis vajuta P ja vali seejärel soovitud variant.

Valikud



Suzanne kenasti eraldatud

Selected (valitud)

Teeb valimist eraldi objekti.

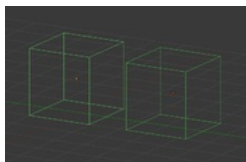
All Loose Parts (kõik eraldi elemendid)

Eraldab objekti mitteühendatud osad.

By Material (materjali alusel)

Jaotab objekti eraldi objektideks vastavalt sellele, mis materjal mingis kohas on.

Objektide grupeerimine



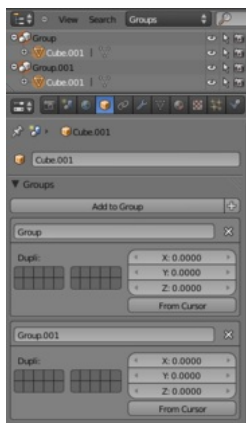
Grupeeritud objektid

Grupeerimine lihtsalt grupeerib, ilma muutusi tekitavate seosteta. Grupeerimist saab kasutada selleks, et asjad oleks paremini organiseeritud, või siis selleks, et ühekorraga terve grupp objekte teisest failist või stseenist sisse tuua või linkida. Gruppi kuuluvate objektide serv on valituna alati roheliselt tähistatud; vaata pilti (*Grupeeritud objektid*).

Valikud

Grupi tekitamine

CtrlG tekitab uue grupi ja paneb sinna valikus olevad objektid.



Grupile nime andmine

Grupile nime andmine

Kui valitud objekt kuulub mingisse gruppi, saab grupi nime näha ja muuta objekti omaduste paneeli Object Properties alluvussuhete paneelil Relations. Nime muutmiseks klõpsa grupi nime väljal.

Struktuuriredaktori (Outliner) aknas nime muutmiseks leia sealt vastav grupp (lülita leidmise lihtsustamiseks kuvamise tüübiks Group) ja tee grupi nimel Ctrl LMB. Nimi muutub muudetavaks tekstiväljaks; muuda nimi ära ja vajuta klahvi ↵ Enter.

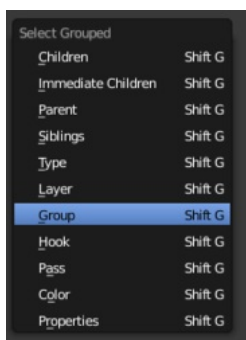
Kihtide abil gruppide sisule piirangute seadmine

Kihinuppude kogum iga grupi juures määrab ära, millistel kihtidel olevaid objekte grupi duplitseerimisel kopeeritakse. Kui grupp sisaldab objekte kihil 10, 11 ja 12, aga näidatud kohast on kiht 12 välja lülitatud, duplitseeritakse ainult kihtidel 10 ja 11 oleva grupi objekte (vaata ka [Dupligruppide](#) võimalusi).

Gruppide sissetoomine ja linkimine

Grupi sissetoomise kohta teisest .blend-failist vaata [seda lehekülge](#). Lühidalt, File → Append or Link → failinimi → Group → grupi nimi.

Grupeeritud objektide valimine



Grupeeritud objektide
valimise hüpikmenüü
Selected Grouped

⇧ ShiftG toob ette dialoogiakna, kus saab valikut teostada alluvussuhete või grupeerimise järgi.

Valikud

Children (alluvad)

Valib kõik valitud objektile alluvad objektid ja omakorda nendele alluvad objektid, kuni nende kõige-kõige viimase alluvani välja.

Immediate Children (otsesed alluvad)

Teeb valituks otsesed alluvad, kuid ei vali nende alluvaid.

Parent (ülem)

Valib objekti ülemobjekti ja samas muudab mittevalituks alguses valitud olnud objekti.

Siblings (võrdsed)

Valib kõik sama ülema alluvuses olevad, aga samal alluvustasemel olevad objektid.

Type (tüüp)

Teeb valituks kõik sama tüüpi objektid.

Layer (kiht)

Teeb valituks kõik aktiivse objektiga samal kihil olevad objektid

Group (grupp)

Teeb valituks kõik objektiga samasse gruppi kuuluvad objektid.

Hook (haak)

Teeb valituks kõik [haakide](#) abil aktiivse objektiga ühendatud objektid.

Pass (renderuskäik)

Teeb valituks kõik objektid, mille renderuskäigu kood (*PassIndex*) on sama, mis aktiivsel objektil. Lisainformatsiooni saamiseks vaata teemat [ID Maski komposiitmissõlme kasutamine](#).

Color (värv)

Teeb valituks kõik objektiga sama värvi olevad objektid (praegusel juhul on tegemist värvusega, mis on määratud paneelil Draw).

Properties (omadused)

Teeb valituks objektid, millele on samasugused [mängumootori](#) omadused.

Järgimine (*Tracking*)

Mode: Objektirežiim Object

Panel: Object » Constraints

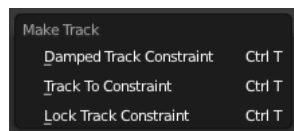
Hotkey: CtrlT

Menu: Object » Track » Make Track

Kirjeldus

Järgimine (*tracking*) tähendab seda, et üks objekt järgib teist objekti. Järgija on *Tracker* ja järgitav on sihtmärk ehk inglise keeles *Target*. Kui järgitav muudab asukohta, siis järgija keerab ennast tema suunas; kui järgija ise muudab asukohta, siis keerab ta ikkagi ennast järgitava suunaliseks. Mõlemal juhul säilitab järgija suuna järgitavale.

Erinevad järgimise variandid



Järgimise piiraja menüü Make Track.

Et panna üks või mitu objekti mingit teist järgima, vali järgivad objektid ja viimasena vali järgitav. Siis vajuta CtrlT. Viimasena valitud objektist saab järgitav ja teistest saavad järgijad (*trackers*). (*Järgimise piiraja menüü Make Track*) abil saab teha mitut tüüpi järgimist:

Teljepiiranguga järgimine (*Track To Constraint*)

Selle abil saab üht objekti teise poole suunata nii, et suunatav jääb alati ühe määratud teljega järgitava poole (*To*) ja teisega globaalse Z-telje poole (*Up*). Järgijat ei saa suunataval teljel pöörata järgijat ennast keerates. Vaata teemat: [järgimise piiraja \(Track To Constraint\)](#)

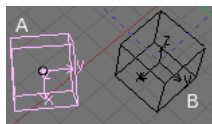
Lukustatud järgimine (*Locked Track Constraint*)

Selle seletamine on keerukas nii sõnas kui ka pildis. See on eelnevaga sarnane, aga siin saab määratud teljel pööramise ära keelata. Ühte objekti saab küll panna ennast teise suunas keerama, aga see toimub ainult ühel tasandil. Näiteks nii, et XY-teljel jälitatava objekti muutmine muudab ka jälitaja suunda, aga Z-teljel nihutamine ei mõjuta. Vaata teemat: [lukustatud järgimine \(Locked Track Constraint\)](#)

Lihtne järgimine (*Damped Track Constraint*)

See suunab ühe järgija lokaalse telje järgitava poole. Vaata teemat: [lihtne järgimine \(Damped Track Constraint\)](#)

Vana järgimise piiraja (*Old Track*)

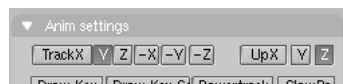


Vana järgimise piiraja (*Old Track* "constraint").


See oli olemas versioonides 2.30 ja vanemates. See on sarnane teljepiiranguga järgimise (Track To) variandile ilma ühtegi telge lukustamata. See püüab säilitada ühe (valik *To*) telje suunamist järgitavale. Kui seda piirajat rakendada, lõppeb see esialgu tavaliselt imeliku suunamisega. Korrekse tulemuse saamiseks kasuta klahvikombinatsiooni AltR. Soovitav meetod on siiski teljepiiranguga (Track To) piiraja kasutamine.

Oletame, et menüüst oli kahe kuubiku märkimise järel valitud siiski vana Old Track. Vaikimisi toimib järgimine nii, et mitteaktiivne objekt on aktiivse poole suunatud +Y teljega. Kuubik "A" järgib kuubikut "B" kasutades Blenderi vana tüüpi (Old Track) piirajat. Nagu näha, on "A" +Y telg suunatud "B" poole, aga orientatsioon on imelik. Tavaliselt on see selle tulemus, et objekti ennast on juba eelnevalt mingil teljel keeratud. Sellisel juhul saab asja korrigeerida nii, et teed järgiva objekti aktiivseks ja vajutad AltR pööramise tühistamiseks.

Järgiva objekti orientatsioon on nüüd selline, et tema enda "Üles"-telg on nüüd ka stseeni ülespoole suunatud telg.



Järgiva telje (*track axis*) määramine.

Selle muutmiseks võta ette animatsiooni sätet paneeli Anim settings, läbi mille saab muuta vana järgimise tüüpi (Old Track) valikuid. Esimesena tee märgituks järgiv objekt (mitte aga sihtmärk) ja muuda nuppude aken Button objekti kontekstiks Object, klõpsates ikoonil () või vajutades F7; vaata pilti (*Järgiva telje (track axis) määramine*).

Seejärel on võimalik järgiva telje (*Tracking axis*) valimine esimeselt nupurealt ja ülespoole suunatud (*upward-pointing*) telje valimine

teisest reast animatsiooni sätete (Anim Setting) paneelilt. Iga kord pärast *Up* telje muutmist rakenda AltR klahvikombinatsiooni. Vastasel korral toimub järgimine vana suunaga. See on üks vana järgimise (Old Track) kasutamise puudusi.

Selle vana järgimise eemaldamiseks märgi ära jälitav objekt ja kasuta klahvikombinatsiooni AltT. Nii nagu varasemate näidete korral, pead ka nüüd valima, kas järgimisest tingitud pööramine säilitada või taastada algne suund.

Märkus

AltT toimib ainult vana järgimise (Old Track) puhul. Tavalise teljepiiranguga piiraja (Track To) ja lukustatud piiraja (Locked Track) eemaldamiseks kustuta need piirajate (Constraints) saki alt.

Vihjed

Aktiivsest objektist saab alati järgitav objekt. Järgimise olemasolu kahe objekti vahel näitab sinine katkendjoon, v.a vana tüüpi järgimise (Old Track) puhul. Kui on näha, et üks objekt järgib teist ilma seda tegevust indikeeriva jooneta, on tegemist vana tüüpi (Old Track) piirajaga.

Mittesobiv järgimine või seadistused

Kui sa valid järgimiseks mittesobiva järgiva (*To*) ja/või üles suunatud (*Up*) telje, jätab järgiv objekt oma suuna alles ja ignoreerib tehtud ebasobivat valikut. Näiteks ei saa korraga teha sellist valikut, et Z-telg oleks teise objekti peale suunatud ja samal ajal oleks Z ülespoole suunatud telg.

Kui järgiva ja üles suunatud telgedega on probleeme, võib sisse lülitada objekti lokaalsete telgede kuvamise. Selleks tuleb Draw paneelil vajutada nuppu Axis (telgede kuvamine). Täpsemalt vaata teemat [kasutajaliides](#).

Duplitseerimine (*Duplication*)

Mode: Muutmiserežiim Edit ja objektirežiim Object

Hotkey: ⇧ ShiftD

Menu: Object → Duplicate

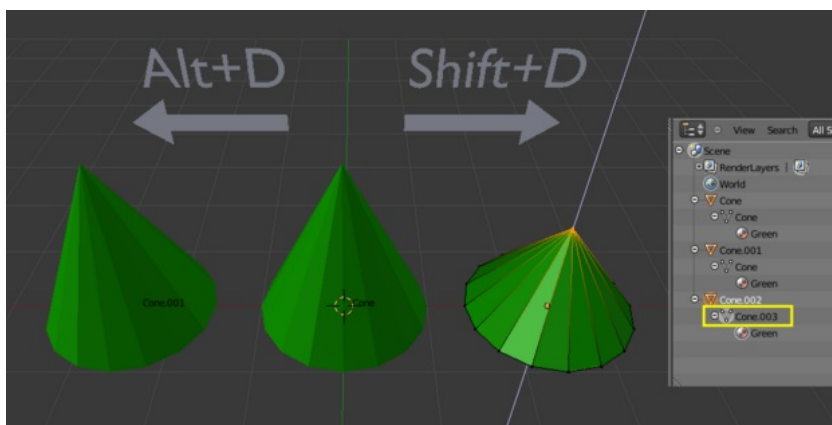
Kirjeldus

See teeb valitud objektist või objektidest täpselt samasuguse väljanägemisega koopia. Koopia tekib alul originaaliga täpselt samasse kohta ja seda saab kohe lohista hakata. Teemakohast näidet vaata altpoolt.

Tekitatud koopia jagab originaaliga vaikselt mõningaid omadusi nagu materjalid, tekstuurid ja lpo-d, kuid näiteks võre on nüüd uus. Seda kopeerimise viisi võib siis nimetada osaliselt lingitud koopiaks. Mõningad osad on lingitud, aga mõningad osad on täiesti uued.

Seda, millised andmeid kopeeritakse ja millised lingitakse, saad määrata kasutaja eelistuste (User Preferences, avaneb File menüüst) muutmise seadete Editing "saki alt" - sealt saad määrata, millised andmed lingitakse ja millised päris eraldi koopiaks muudetakse.

Näited



Toimub objekti Cone.002 võre Cone.006 muutmise. Struktuurivaates (Outliner) on parajasti töödeldav unikaalne andmeblokk kuvatud heledamalt.

Keskel olevat koonust on kaks korda duplitseeritud. Vasemal olev on lingitud duplikaat ja paremal linkimata.

- Parempoolset koonust muutes keskmine koonus ei muutu. Võre andmed ei ole lingitud, vaid kopeeritud.
- Samamoodi ei juhtu keskmise koonusega midagi siis, kui parempoolset koonust objektirežiimis muuta. Nii et uue objekti puhul on ka teisendamise omadused kopeerimisel iseseisvaks muutunud.
- Parempoolse duplikaadi tegemisel on säilinud keskmisega sama materjal. Materjali omadused ei ole kopeeritud, vaid lingitud.

Kui sa soovid vaikselt lingitud andmetest iseseisvat koopiat teha, siis loe edasi.

Lingitud duplikaadid (*Linked Duplicates*)

Mode: Objektirežiim Object

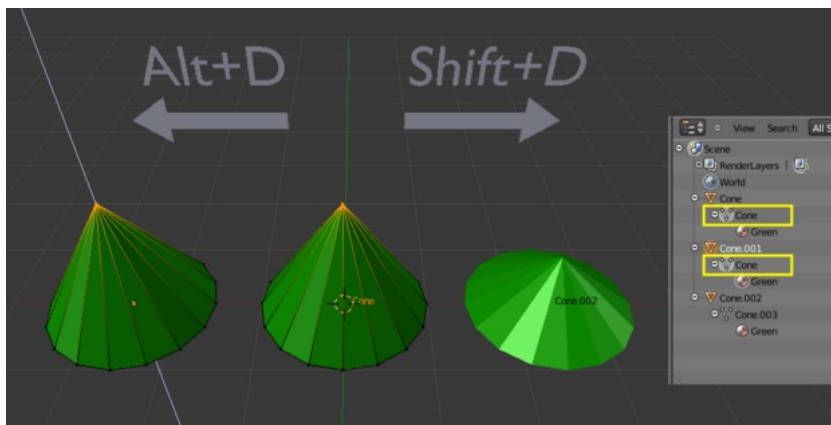
Hotkey: AltD

Menu: Object → Duplicate Linked

Kirjeldus

Päris duplikaadi asemel saab teha ka lingitud duplikaadi. Kõik uue objekti andmed on siis originaaliga lingitud. Kui ühte objekti muutmise režiimis Edit muuta, rakenduvad kõik muudatused ka lingitud koopiatele. Teisendamise omadused on siiski igal objektil unikaalsed, nii et objektirežiimis tehtud keeramised, mõõtkava teisendamised ja nihutamised teisi koopiaid ei mõjuta. Teemakohast näidet vaata altpoolt.

Näited



Objekt Cone.001 on lingitud duplikaat. Kuigi neil on nüüd unikaalsed nimed, on nende võred lingitud (struktuurivaates esiletõstetud võre nimega Cone).

Vasempoolne on keskmise lingitud duplikaat (kasutati klahvikombinatsiooni AltD).

- Kui muutmiserežiimis üht tippu liigutada, liigub sama tipp ka originaalobjektile. Võre andmed on kopeerimise asemel lingitud.
- Objektirežiimis tehtavad muudatused aga originaali ei mõjuta - need andmed on unikaalsed. Teisendamise omadused on linkimise asemel kopeeritud.
- Nii nagu eelmise näite puhul, on ka nüüd originaalobjekti ja koopia materjal ühine. Materjali omadusi pole kopeeritud, vaid need on lingitud.

Tavaline laud koosneb lauaplaadist ja neljast jalast. Tee üks jalg valmis ja siis tee 3 korda lingitud koopia. Tulemuseks on see, et üht jalga muutes muutuvad kõik korraga. Lingitud duplikaate võib kasutada ka näiteks joogiklaaside, autorataste ja muude korduvate või sümmeetriliselt paigutatud objektide korral.

Protseduuriline duplitseerimine (*Procedural Duplication*)

Mode: Muutmiserežiim Edit ja objektirežiim Object

Panel: Object settings

Protseduurilise koopja tegemiseks on Blenderis praegusel hetkel neli varianti. Need saab kätte objektimenüüst Object.

Tipud (Verts)

See teeb iga objekti tipu kohta objekti kõigist alamobjektidest sellesse tippu paigutatud koopja (toimib ainult võreobjekti puhul).

Küljed (Faces)

See teeb iga objekti külje kohta objekti kõigist alamobjektidest selle külje keskele paigutatud koopja (toimib ainult võreobjekti puhul).

Grupp (Group)

Teeb grupist koopja, mis pärib grupi ülemobjekti teisendused. Duplitseeritud gruppi saab tegevuste (Actions) abil animeerida. Võib kasutada ka [proksiobjekti](#).

Kaadrid (Frames)

Animeeritava objekti korral teeb see uue "koopja" iga kaadri kohta. Hiljem näed, et see on väga võimas vahend objektide paigutamisel ja modelleerimisel.

Lingitud andmekogud (*Linked Library Duplication*)

Hotkey: ⇧ ShiftF1

Menu: File → Link Append

Lingitud teegid

Teisest failist andmete linkimine on veel üks duplitseerimise meetod. Igat välises .blend failis olevat andmeblokki saab käsilolevas failis taaskasutada.

Soovitused

Kui sa tahad, et ka teisendusomadused oleksid lingitud, vaata peatükki [Objektide grupeerimine ja allutamine](#).

DupliTipud (*DupliVerts*)

Mode: Objektirežiim Object

Panel: Object → Duplication

DupliTipud Duplication Vertices ehk DupliVerts tähistavad aluseks oleva objekti duplitseerimist mõne võre tippude asukohas. Teiste sõnadega, kasutades DupliTippe DupliVerts mõne võre peal, asetatakse võre igasse tippu alusobjekti esindaja.

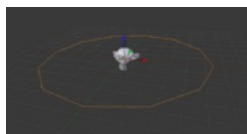
DupliTippudega saab tegelikult modelleerida kahel viisil. Neid saab kasutada paigutamise tööriistana, mis lubab meil modelleerida objektide jaoks geomeetrilist paigutust (nt Kreeka templi sambad, puud aias, robotsõdalaste armee, klass koolipinkidega). Objektid võivad olla ükskõik millisest objekti tüübist, mida Blender toetab. Teine võimalus on kasutada neid selleks, et modelleerida objekte, alustades nende ühest osast (nt nua ogad, merisiiliku okkad, seina plaadid, õie kroonlehed).

Lae alla .blend-näidisfail

Siit saad alla laadida sellel lehel olevaid näiteid sisaldava faili. [Selles .blend-failis](#) on esimene näide (ringile allutatud pärdik) kihil 1; kihil 2 on ikosaeedrile allutatud kombits. Failid tehti Blenderi versioonis 2.55.1 (r33567).

DupliTipud kui paigutamise tööriist

Alustamine



Pärdiku pea ja ring

Vajad ainult alusobjekti (nt *puu* või *sammas*) ning võret, mille tipud vastaksid sellele mustriks, mida soovid. Selles osas kasutame ülejäänud sammude jaoks lihtsat stseeni. Kasutame alusobjektina koordinaatide algpunktis asuvat pärdikupead ning meie ülemobjektiks ehk paigutamist määravaks võreks on samas kohas asuv ring.



DupliTippudega
paigutatud pärdik

Kõigepealt vali objektirežiimis alusobjekt ning vajuta Shift RMB , et lisada valikusse ring (järjekord on siinkohal väga oluline) ning seejärel CtrlP, et allutada alusobjekt ringile. Nüüd on ring pärdiku ülem; kui liigutad ringi, liigub pärdik koos sellega.

Valides ainult ringi, lülita sisse DupliTipud Duplication vertices paneelis Object → Duplication → Verts. Igasse ringi tippu peaks ilmuma üks pärdikupea.

Algset pärdikupead keskpunktis ja ülemvõret näidatakse 3D-vaates edasi, kuid kumbagi neist ei renderdata. Kui ahvi pea asukoht ja pööre on imelikud, võid soovida tühistada selle pöörde (AltR), mõõtkava AltS, asukoha AltG ja keskpunkti AltO.

Ümberpaigutamine

Kui sa nüüd valid alusobjekti ning muudad seda objekti- või muutmisrežiimis, mõjutavad kõik muudatused ka duplitseeritud objekti kuju. Samuti võid valida ülemvõre, et muuta duplikaatide paigutust; tippude lisamisel lisanduvad ka uued alusobjektid. Pane tähele, et alusobjektid omandavad muudatused, mis tehti ülemvõrele objektirežiimis, kuid mitte neid, mis tehti muutmisrežiimis; objektirežiimis ringi mõõtkava suurendamine suurendab ka pärdikupead, aga ringi suurendamine muutmisrežiimis suurendab ainult alusobjektide vahelist kaugust.

Suunamine



Sisselülitatud suunamine
Orientation, suund +Y

Alusobjektide suunda saab muuta, kui lülitada sisse Duplication paneelis asuv pöörde valik Rotation. See pöörab kõiki alusobjekte vastavalt ülemvõre tipunormaalidele.

Et muuta duplitseeritud objektide suunda, vali alusobjekt ning muuda paneelis Object → Animation Hacks järgimistele valikut Tracking Axes.

Erinevate suundade tulemus



Negatiivne Y

Positiivne X

Positiivne Z, X üleval

Märkus

Objekti telgesid saab muuta nähtavaks paneelist Object→ Display.

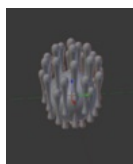
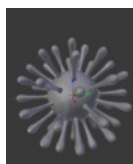
Et näha ülemvõre tipunormaale, sisene tabulaatoriga muutmisrežiimi ning lülita see võimalus sisse paneelis Properties (N)→ Display, kust saad ka vastavalt vajadusele muuta kuvatavate normaalide suurust.

DupliTipud modelleerimise tööriistana

Kasutades DupliTippe ja tavalisi primitive, saab luua väga huvitavaid mudeleid. Selles näites loodi lihtne kombits, eendades paar korda kuupi. See kombits allutati seejärel ikosaeederkerale. Ülemvõre (ikosaeederkera) puhul lülitati sisse duplitseerimise pööre (Rotation) ning alusobjektide (kombitsad) suund vastas ülemvõre tipunormaalidele (antud juhul pöörati kombitsat muutmisrežiimis - 90° mööda X-telge).



Lihtne silutud kombits

Ülemvõrele
duplitipuna allutatud
kombitsSisselülitatud
pööramine, et
duplikaate joondada

Nagu eelmise näitegi puhul, saab ka nüüd paigutuse kuju ja propotsioone edasi sättida.

Et muuta kõiki duplikaate reaalseteks objektideks, vali lihtsalt ikosaeederkera ning seejärel Object→ Apply→ Make Duplicates Real (Ctrl⇧ ShiftA). Et muuta ikosaeederkera ja kombitsad üheks objektiks, vali nad kõik ja kasuta menüüvalikut Object→ Join (CtrlJ).

Loe edasi

Teised duplitseerimise meetodid on esitatud [siin](#).

DupliKüljed (*DupliFaces*)

Mode: Objektirežiim Object

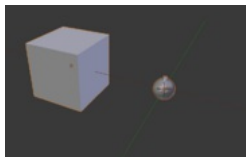
Panel: Object → Duplication

Duplitseeritud küljed (Duplication Faces) või lühidalt dupliküljed (DupliFaces) on võimalus objekte kopeerida nii, et koopia tekib iga ülemobjekti külje kohta. Kõige parem on seda seletada joonise abil.



Näidis .blend failina

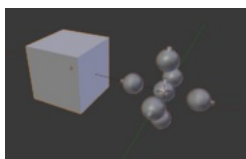
Tõmba endale näidiseks alla [see fail](#)

Asja põhimõte



Kuubik ja kera

Selles näites kasutame me alamobjektina UV-kera, mille "põhjapoolus" on pisut eendatud, ja ülemobjektina kuubikut. Kera allutamiseks kuubikule tuleb esmalt olla objektirežiimis, siis valida klahviga RMB  kera ja seejärel valida ⇧ Shift RMB  kombinatsiooniga kuubik (valimise järjekord on oluline). Lõpuks tuleb allutamiseks vajutada klahvikombinatsiooni CtrlP.



Dupliküljed kuubikule omistatuna

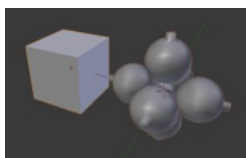
Objektikontekstis Object tuleb duplikatsiooni paneelilt Duplication sisse lülitada külgede valik Faces. Tulemuseks on see, et kerast tehakse kuubiku iga külje kohta koopia.

Päritud omadused

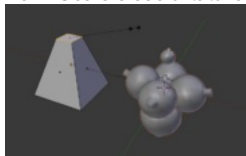
Asukoht, suund ja suurendus sõltuvad kõik ülemobjekti külgede vastavatest omadustest. Kui kuubikule on allutatud mitu objekti, tekib neist kõigist üks koopia ühe kuubiku külje kohta. Kui nüüd kuubiku võret muutmisrežiimis tükeldada (Subdivide, W), tekib iga uue külje kohta ka uus kera.

Nii ülemobjekt kui ka originaal on 3D-vaates nähtavad ja muudetavad, kuid neid ei renderdata.

Mõõtkava (*Scale*)



Valik Scale sisselülitatuna



Kuubiku ülemist külge on vähendatud

Mõõtkava mõju valiku Scale sisselülitamisel pärivad alamobjekti koopiad ülemobjekti küljelt selle suhtelised mõõtmed.

Ülemobjekti külje suurendamisel suurenevad vastavalt ka sellele duplitseeritud objektid.

DupliGrupp (*DupliGroup*)

Mode: Objektirežiim Object

Panel: Object → Duplication → Group

Duplikatsioonigrupp Duplication Group ehk DupliGroup võimaldab sul luua grupi, mis koosneb mingisuguse objekti kõigist esindajatest.

Üldine kasutamine

- Loo hunnik objekte ja grupeeri nad
 1. valides nad kõik,
 2. vajutades CtrlG ning
 3. muuda grupi nimi hiljem paneelis Object → Groups
- Loo DupliGrupp
 1. lisades uue objekti (⇧ ShiftA), näiteks Empty,
 2. paneelis Object → Duplication lülita sisse grupisäte Group
 3. ning leia ilmuvast valikukastist äsjaloodud grupi nimi.

DupliGrupid ja dünaamiline linkimine

Loe [lisamise ja linkimise peatükki](#), et mõista, kuidas dünaamiliselt mõnest teisest .blend-failist andmeid parajasti avatud faili linkida. Dünaamiliselt saab ühest .blend-failist linkida teise ka gruppe. Kui sa seda teed, ei ilmu lingitud grupp stseeni, enne kui sa lood objekti, mis määrab, kus kohas grupi esindaja olema peab.

Näide

- Lingi mõnest teisest failist enda stseeni mõni grupp, nagu seda kirjeldatakse [lisamise ja linkimise peatükis](#). Seejärel saad valida lihtsa või raske tee:
- Lihtne tee:
 1. Vali Add → Group Instance → [äsjaloodud grupi nimi].
- Raske tee:
 1. Vali Add → Empty ning vali seejärel äsja lisatud tühiobjekt.
 2. Mine objekti konteksti Object duplikatsiooni paneeli Duplication ja vajuta grupi nupule Group.
 3. Grupi Group: kõrvale ilmuvast rippmenüüst vali enda poolt lingitud grupp.

Koheselt ilmub ka selle grupi esindaja. Tühiobjekti võid duplitseerida ning iga duplikaadi puhul säilitatakse ka DupliGrupi sätted. Sedasi saad väga lihtsalt luua lingitud andmetest mitmeid koopiaid.

DupliGrupi objekti reaalseks muutmine

Oletame, et sa soovid DupliGrupi muuta või renderdada seda Yafaray's või mõnes muus renderdajas, mis ei toeta otsest DupliGruppide importimist:

Vali lihtsalt DupliGrupp ja vajuta Ctrl⇧ ShiftA, et see tavaliseks objektiks muuta, mida saab siis tavapäraselt teisendada ja animeerida.

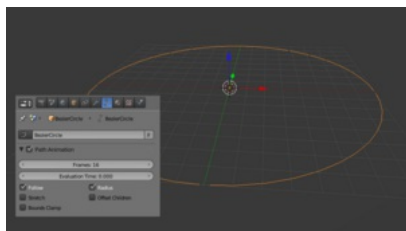
Märkus

Pea meeles, et kui DupliGrupp lingiti mõnest välisest failist, siis objekti andmed (võre, materjalid, tekstuurid, teisendused) on samuti lingitud algsest grupist. Kuid erinevaid objekti ülem-alamsuhteid üle ei tooda.

DupliKaadrid (*DupliFrames*)

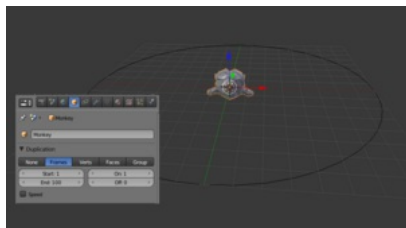
DupliKaadrite abil saab objekte duplitseerida mööda kõverat asuvates kaadrites. Nii saab kiiresti objekte paigutada.

Näited



Kõvera seaded

Lisa ⇧ ShiftA abil Bezier' ring (Bezier Circle) ja suurenda seda veidi. Lülita Curve menüüst Path Animation seksioonis sisse Follow (kõvera jälitamine) ja pane Frames (kaadrid) väljale midagi mõistlikumat kui 100. Näiteks 16.

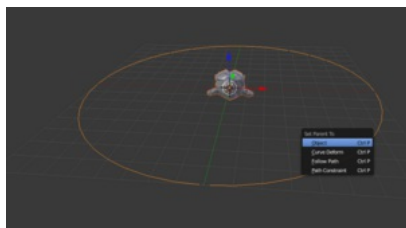


Objekti seadistamine

Lisa pärdikobjekt (Monkey). Olles objekti menüü Object duplikatsiooni seksioonis Duplication, lülita sisse Frames (kaadrid) ja lülita välja Speed (kiirus).

Kiirus (*Speed*)

Seda kasutatakse siis, kui objekt on alluvussuhte abil määratud liikuma mööda kõverat (Follow Path). Toodud näites liigub pärdik mööda kõverat 16 kaadri jooksul.



Allutamine (*Parenting*)

Objekti allutamiseks kõverale vali kõigepealt pärdik ja seejärel lisa valikusse (Shift-klahvi all hoides) kõver ning vajuta CtrlP. Vali pärdik ja vajuta AltO, et viia pärdiku keskpunkt koordinaattelgede nullpunkti.



Suuna muutmine

Nüüd saad pärdiku suunda muuta seda keerates (nii muutmise kui ka objekti režiimis) või kui muudad raja jälitamise peatelge (Tracking Axes) Animation Hacks seksiooni alt (pärdik peab olema valitud). Pärdikute paigutust saab muidugi muuta ka kõvera muutmise abil.

Pärdikute pärisobjektideks muutmiseks vajuta Ctrl⇧ ShiftA või vali menüüst Object→Apply→Make Duplicates Real. Kõik pärdikud on nüüd päris objektid, kuid need on siiski lingitud originaalpärdikuga. Selle muutmiseks võta menüüst Object→Make Single User→Object&Data ja siis vali All.

Märkus

DupliKaadritele on olemas mitmed alternatiivid. Millist tööriista kasutada, sõltub vajadusest.

1. Pisikese kõvera profiilina ja suurema kõvera rajana kasutamiseks lisa esimene teisele kantimise objektina (Bevel Object).
2. Objektide paigutamiseks kõverale on hea kasutada reatöötlejat (Array Modifier) ja kõveratöötlejat (Curve Modifier).
3. Duplivers ehk DupliTipud on kasulikud näiteks objekti paigutamiseks mööda ringjoont või tükeldatud pinda.

Välised viited

- [Blender Artists: *Dupliframes in 2.5*](#)

Muutmisrežiim (*Edit Mode*)

Muutmisrežiimi sisenemine

Geomeetriliste objektidega toimetamiseks on olemas kaks režiimi.

Objektirežiim (Object mode)

Objektirežiimis (Object Mode) tehtavad muudatused muudavad tervet objekti.

Objektirežiimis on 3D-vaate aknas selline päis:



Objektirežiimi (Object Mode) akna päis

Muutmisrežiim (Edit mode)

Muutmisrežiimis tehtud toimetused muudavad ainult objekti geomeetriat, aga ei mõjuta selliseid omadusi nagu objekti asukoht ja kaldenurk.

Muutmisrežiimi korral on 3D-akna allservas selline vaade:



Muutmisrežiimi (Edit Mode) päis

Tööriistad ja valikud 3D-vaate päises muutmisrežiimi korral (vasakult paremale):

View(kuvamine), *Select* (valikud), *Mesh* (võre) menüüd

Blenderi üldrežiim

3D-vaate kuvamisrežiim

Keskpunkti (*pivot*) seadistus

3D-manipulaatori seadistus

Valiku tegemise režiim

Sügavuspühvri piiraja seadistus

Proportsionaalse muutmise seadistus

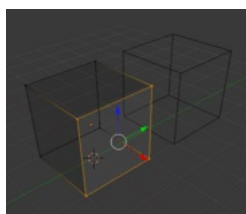
Nakkumine

OpenGL-i kiirrenderdus

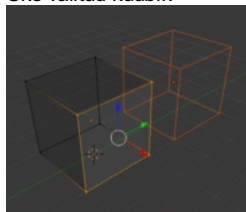
Nende kahe režiimi vahel saab liikuda klahviga \leftrightarrow Tab. Sama saab teha ka hiirega, kui 3D-vaate alaserva ribal valida hüpikmenüüst Mode vajalik režiim.

Pärast objekti tekitamist võid sa koheselt sattuda muutmisrežiimi – see sõltub sellest, kas muutmisrežiimi sisenemise nupp Switch to Edit Mode on kasutaja eelistustes (User Preferences) saki Editing alt sisse lülitatud või mitte. Muutmisrežiim mõjutab korraga ainult ühte objekti, milleks on valitud või viimati valitud olnud objekt.

Visualiseerimine



Üks valitud kuubik



Kaks valitud kuubikut enne muutmisrežiimi minekut

Vaikeseadistuse korral näitab Blender valitud objektide ümber oranži äärejoont. Seda nii objektirežiimi kui ka muutmisrežiimi korral. Värv saab muuta kasutaja eelistuste kaudu (User Preferences, Ctrl+Alt+U →Themes).

Objektirežiimis traatvõre (Wireframe) varjutuse sisselülitamisel (Z) näidatakse mittevalitud objekte musta värviga ja valitud objekte oranži värviga. Mitme objekti valimise puhul näidatakse kõiki teisi peale viimase pisut tumedama oranži värviga. Nii on see ka muutmisrežiimis: mittevalitud geomeetriat näidatakse mustaga, kuid valitud küljed, servad ja tipud on kuvatud oranžina. Aktiivne külg kuvatakse heledamalt.

Muutmisrežiimis võib korraga olla vaid üks võre. Siiski võib mitu objekti objektirežiimis üheks kokku ühendada (Ctrl+J) ja seejärel muutmisrežiimis lahutada (P). Kui muutmisrežiimi minnes on valitud mitu objekti, jäävad kõikide valitud objektide servad oranžideks. Nii on jätkuvalt näha, et need kuuluvad valikusse.

Kui muutmisrežiimi tippude valimise režiimis (Vertex selection mode) on valitud kaks tippu, muutub valituks ka neid ühendav serv. Samaselt sellele muutub valituks terve külg, kui on valitud kõik selle külje tipud.

Tööriistariiul (*Tool Shelf*)



Muutmisrežiimi
tööriistariiuli paneel
(paneel on kaheks
jaotatud
küljenduspõhjustel)

Võre tööriistade (Mesh Tools) paneeli avamiseks/sulgemiseks kasuta klahvi T. Pärast muutmisrežiimi sisenemist ilmuvad nähtavale mitmed võre töötlemiseks mõeldud tööriistad.

Peaaegu kõigi riistade jaoks on olemas klahvikombinatsioonid (kuvatakse näpunäitena iga riista korral), lisaks on need saadaval erivalikute (Specials) menüüs, vajutades klahvi W, serva tööriistade menüüs (Edge, CtrlE) ja külje tööriistade menüüs (Face, CtrlF). Iga riista korral tekib tööriistariiulisse kontekstipõhine nuppude valik.

Veel rohkem võretööriistu saab sisse lülitada, kui minna kasutaja eelistuste seadetes (User Preferences) ja sealt edasi lisatööriistade alajaotusesse (Add-Ons). Uusi riistu arendatakse pidevalt juurde ja edasi. Nende kohta saab infot Blenderiga seonduvatelt veebilehtedelt ja foorumitest.

Täiendava informatsiooni saamiseks vaata [lisatööriistade paneeli](#) teemat.

Omaduste ala (*Properties Shelf*)



Muutmisrežiimi Omaduste
ala (Properties Shelf)
(paneel on kaheks
jaotatud
küljenduspõhjustel)

Selle avamiseks/sulgemiseks vajuta klahvi N.

Omaduste alal on otseselt võre muutmisega seotud teisenduste paneel (Transform), kus saab numbrilisi väärtusi sisestada, ja võre kuva seadete (Mesh Display) paneel. Selle kaudu saab näiteks sisse lülitada normaalide kuvamise, külgede pindala kuvamise ja servade omavahelise nurga kuvamise.

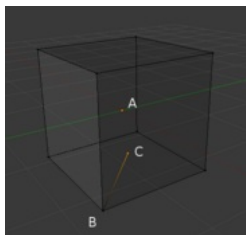
Mitmeid tööriistu leiab veel objekti andmete (Object Data) paneelilt, nagu näiteks erinevad kuvamise seaded, tipugrupid (Vertex groups) ja muud.

Täiendava informatsiooni saamiseks vaata [lisatööriistade paneeli](#) teemat.

Tipud, servad ja küljed


Tavalistes võreobjektides koosneb kogu võre kolmest põhilisest struktuurielemendist: tipud, servad ja küljed (me ei räägi praegu kõveratest, NURBSidest ja muudest objektitüüpidest) Kuid pettuda pole tarvis: need lihtsad elemendid annavad meile rikkalikud võimalused mudelite loomiseks.

Tipud (*Vertices*)



Tipu näidis

Tipp on üksik punkt või asukoht 3D-ruumis. Tavaliselt on see renderdamisel ja objektirežiimis mittenähtav. Ära aja omavahel segi objekti keskpunkti ja tippu. Need näevad sarnased välja, kuid keskpunkt on suurem ja seda ei saa aktiivseks teha. (*Tipu näide*) pildil on keskpunkt tähistatud tähega "A". "B" ja "C" on tipud.

Lihtne viis uue tipu tekitamiseks on Ctrl LMB  muutmise režiimis. Kuna arvuti ekraan on 2-mõõtmeline, siis info sügavuse kohta puudub ja uus punkt tekitatakse sama sügavale, kui on parajasti 3D-kursor. Uue tipu tekitamisel ühendatakse see automaatselt serva abil viimasena valikus olnud tippudega. Pildil olev ja C-tähega tähistatud tipp on uus lisatud tipp ja tippude B ja C vahele on tekkinud serv.

Servad (*Edges*)

Serv ühendab omavahel sirge joonega 2 tippu. Servad on need "traadid", mida on näha traatvõre (*wireframe*) vaates. Tavaliselt neid renderdatud pildil näha ei ole Nende abil moodustatakse küljed (*faces*). Serva valmistamiseks tee valituks kaks tippu ja vajuta F.

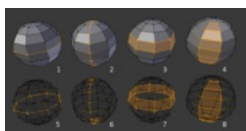
Küljed (*Faces*)

Külg on võreobjekti kõige kõrgema taseme struktuur. Külgede abil ehitatakse objekti tegelik pind. Neid renderdamisel näidataksegi. Külje saab moodustada otsapidi ühendatud servade vahele (kolmnurk) või nelja tipu vahele: sel juhul on tulemuseks servade abil ühendatud nelinurk. Kolmnurgad on alati lameded ja sellest tulenevalt on nendega opereerimine lihtne. Kuid nelinurki on jälle parem deformeerida ja tükeldada ning seetõttu on see modelleerimisel eelistatum.

Ole nelja küljega külje suhtes ettevaatlik, sest sisemiselt töödeldakse neid kahe ühendatud kolmnurgana. Nelinurkne külg toimib hästi ainult siis, kui see on lame (kõik punktid on samas tasapinnas) ja ühegi nurga suurus ei ületa 180 kraadi. Nii on see näiteks kuubikuga. Selle tõttu pole traatvõre vaates diagonaale näha.

Kolmnurkadest koosnevate külgedega kuubik on küll võimalik kuid tulemuseks on segasem pilt muutmisrežiimis. Kolme või nelja külgedega piiratud tipu vahel ei pea alati olema külg. Kui külg puudub, siis renderdatakse see ala tühja kohana. Külje moodustamiseks vali 3 või 4 sobivat tippu ja vajuta F.

Silmused (*Loops*)



Serva- ja küljesilmused

Serva- (Edge) ja küljesilmused (Face Loops) on ridamisi omavahel kokku puutuvatest servadest või külgedest moodustatud jada. Vaata pilti (*Serva- ja küljesilmused*). Ülemises reas olevad (1-4) näitavad samadest silmustest tahke keha vaadet, alumine rida (5-8) aga traatvõre vaadet.

Pane tähele, et silmused 2 ja 4 ei käi ümber terve mudeli. Nende silmuste lõpp on n-ö. poolustel, sest sealt edasiminekuga puudub ühene võimalus. Poolused on tipud, mis ühendavad omavahel kolme, viite või rohkemat serva. Poolused ei ole need tipud, mis ühendavad ainult ühte, kahte või nelja serva.

Pildil kuvatavad poolusi vältivad silmused on tsüklilised (1 and 3). Need algavad samast tipust ja ka lõppevad samas tipus ja need jagavad mudeli kaheks osaks. Silmused on mudelitega töötamisel väga võimsaks vahendiks. Ka animeeritud tegelaste puhul on eelistatud silmustega modelleerimine, kuna see võimaldab puhtamat ja anatoomiliselt korrektsemat modelleerimist. Täpsema info saamiseks vaata teemat [serva ja külje tööriistad](#).

Serva silmused (*Edge Loops*)

Silmused 1 ja 2 pildil (*Serva- ja küljesilmused*) on serva silmused. Tipud on omavahel silmuseks ühendatud, nii et igal silmuses oleval tipul on kaks naabertippu, mis ei kuulu silmusesse (välja arvatud poolustel asuvad algus- ja lõputipud).

Servasilmused on orgaanilisel modelleerimisel ja animeeritud tegelaskujude loomisel väga vajalikud. Kui neid õigesti kasutada, siis läheb tõepäraseks modelleerimiseks tarvis suhteliselt vähe tippe. Selliseid objekte on hea tihedamateks osadeks jagada ja need alluvad hästi deformatsioonidele.

Servasilmustest koosneva mudeli puhul jälgib kontuur paremini lihaste liikumist ja näiteks põlved deformeeruvad liikumisel palju loomulikumalt.

Täpsemat informatsiooni servasilmustega töötamise kohta saad teemast [servasilmuste valimine](#).

Küljesilmused (*Face Loops*)

Küljesilmused on loogiline laiendus servasilmustele ja neid on näha mudelil 3 ja 4 pildil (*Serva- ja küljesilmused*). Näite (4) puhul on näha, et küljesilmuse sisse ei kuulu küljed, mille üks tipp on poolus.

Täpsemat informatsiooni tööks küljesilmustega saad teemast [küljesilmuste valimine](#).

Võre baasobjektid

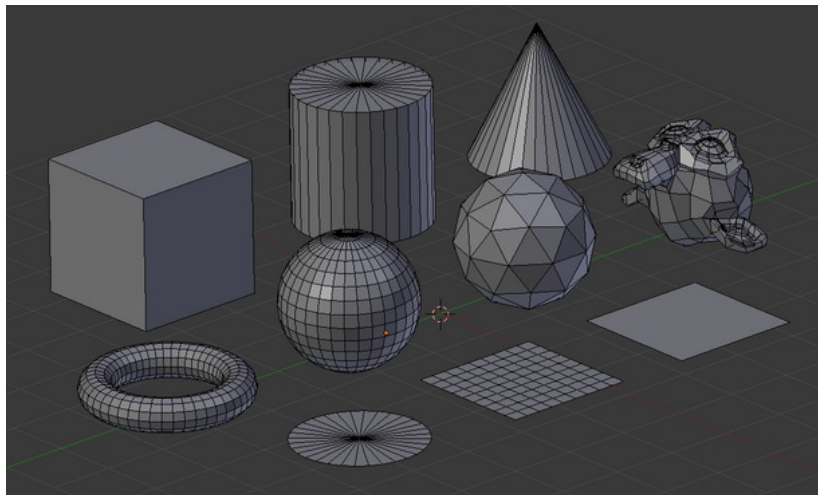
Mode: Objektirežiim Object

Hotkey: \diamond ShiftA

Menu: Add » Mesh

Kirjeldus

Tavapärane 3D-stseenides kasutatav objektitüüp on võre. Blenderiga tuleb kaasa hulk "primitiivseid" võrekujusid, millest saad oma modelleerimist alustada.



Blenderi kümme standardset primitiivi

Rohkem kui ühe primitiivi puhul muudetavad sätted on:

Radius (raadius)

Määrab algsuuruse ringil (Circle), silindril (Cylinder), koonusel (Cone), UV-keral (UVSphere) ja ikokeral (IcoSphere).

Depth (sügavus)

Määrab algpikkuse silindril (Cylinder) ja koonusel (Cone).

Märkus tasapinnaliste primitiivide kohta

Saad tasapinnalise võre muuta kolmemõõtmeliseks, liigutades ühe või enam tippu tasapinnast välja (töötab tasapinna (Plane), ringi (Circle) ja ruudustiku (Grid) puhul). Lihtsat ringi kasutatakse tegelikult sageli ka väga keeruliste võrede alguspunktina.

Valikud

Tasapind (Plane)

Tavalisel tasapinnal on neli tippu, neli serva ja üks külg. See on nagu laual lebav paberitükk; see ei ole reaalne kolmemõõtmeline objekt, sest see on lame ja sellel puudub paksus. Objektid, mida saab tasapindadega luua, on näiteks põrandad, lauaplaadid ja peeglid.

Kuup (Cube)

Tavalisel kuubil on kaheksa tippu, kaksteist serva ja kuus külge ning see on reaalne kolmemõõtmeline objekt. Objektid, mida saab kuubiga luua, on näiteks täringud, karbid ja kastid.

Ring (Circle)

Tavalisel ringil on n tippu. Tippude hulga ja raadiuse saab määrata tööriistariiulis (Tool Shelf) olevas kontekstipaneelis, mis ilmub pärast ringi loomist. Kui täitmise nupp Fill on sisse lülitatud, täidetakse ring kolmnurksete külgedega, mis jagavad keskel ühte tippu. Kuid ring on siiski ainult lame kuju. Kui see ei ole täidetud ja sa soovid seda renderdada, pead määrama sellele materjali menüüst traatvõre materjali. Raadiuse parameeter Radius määrab ringi suuruse.

Mida rohkem tippe ringil on, seda siledam on selle kontuur; vaata pilti (*Erinevate sätetega saadud "ringid"*). Ainult kolme tipuga ring on tegelikult kolmnurk, ring on lihtsalt tavaline moodus, kuidas lisada hulknurki nagu kolmnurgad, viisnurgad jne.

UV-kera (UV Sphere)

Tavalisel UV-keral on n segmenti ja m ringi. Detailitäpsuse ja raadiuse saab määrata tööriistariiulis (Tool Shelf) olevas kontekstipaneelis, mis ilmub pärast UV-kera loomist. Segmentide ja ringide hulga suurendamine muudab UV-kera pinna siledamaks. Segmentid on nagu maakera pooluselt poolusele minevad meridiaanid ning ringid (Rings) nagu maakera paralleelid. Objektid, mida saab UV-keraga luua, on näiteks pallid, pead ja pärlid.

Märkus

Kui sa määrad kuue segmenti ja kuue ringiga UV-kera, siis on tulemuseks miski, mis näeb üldvaates välja nagu kuusnurk (kuus segmenti), millel on viis ringi ning kaks poolustel olevat punkti. Seega kaks ringi vähem kui ootasid või üks ring rohkem, kui loed pooluseid 0-raadiusega ringideks.

Ikokera (Icosphere)

Ikokera on kolmnurkadest loodud kera. Tükelduste (*subdivision*) hulga ja raadiuse saab määrata tööriistariiulis (Tool Shelf) olevas kontekstipaneelis, mis ilmub pärast ikokera loomist; tükelduste arvu suurendamine muudab ikokera siledamaks. Tasemel 1 on ikokera ikosaeder, 20 võrdkülgsest kolmnurgast koonsev tahke objekt. Tükelduste taseme suurendamisel jagatakse iga kolmnurga külg neljaks kolmnurgaks ning tulemuseks on kerajam välimus. Ikokerasid kasutatakse tavaliselt

isotroopsemate ja ökonoomsemate tipupaigutuste saavutamiseks kui UV-kera puhul.

Märkus

Võimalik on lisada 500 korda tükeldatud ikokerasid. Niivõrd tiheda võre lisamine on kindel moodus, kuidas oma arvuti kokku jooksutada. 10 korda tükeldatud ikokeral on 5 242 880 kolmnurka, nii et ole sellega väga ettevaatlik!

Silinder (Cylinder)

Tavalisel silindril on n tippu. Tippude hulga ringikujulises läbilõikes saab määrata tööriistariulis (Tool Shelf) olevas kontekstipaneelis, mis ilmub pärast objekti loomist; mida kõrgem on tippude arv, seda siledam on ringikujuline läbilõige. Raadiuse (Radius) ja sügavuse (Depth) parameetrid määravad silindri mõõdud. Objektid, mida silindritega luua, on näiteks käepidemed ja vardad.

Kui sulgev ots Cap Ends ei ole sisse lülitatud, on tulemuseks toru. Objektid, mida nendest torudest luua saab, on näiteks veetorud ja joogiklaasid (peamine erinevus silindri ja toru vahel on see, et esimese otsad on suletud).

Koonus (Cone)

Tavalisel koonusel on n tippu. Tippude hulga ringikujulises aluses, mõõtmed ja võimaluse koonuse alus sulgeda saab määrata tööriistariulis (Tool Shelf) olevas kontekstipaneelis, mis ilmub pärast objekti loomist; mida kõrgem on tippude arv, seda siledam on ringikujuline alus. Objektid, mida saab koonusega luua, on näiteks ogad ja teravatipulised mütsid.

Toroid (Torus)

Pontšikukujuline primitiiv, mille loomiseks pööratakse ringi ümber telje. Üldised mõõtmed määratakse suure (Major) ja väikese raadiuse (Minor Radius) kaudu. Tippude arv (segmentides) võib mõlemal ringil olla erinev ning see määratakse tööriistariulis (Tool Shelf) olevas kontekstipaneelis kahe raadiuse abil (suured segmendid (Major Segments) ja väikesed segmendid (Minor Segments)).

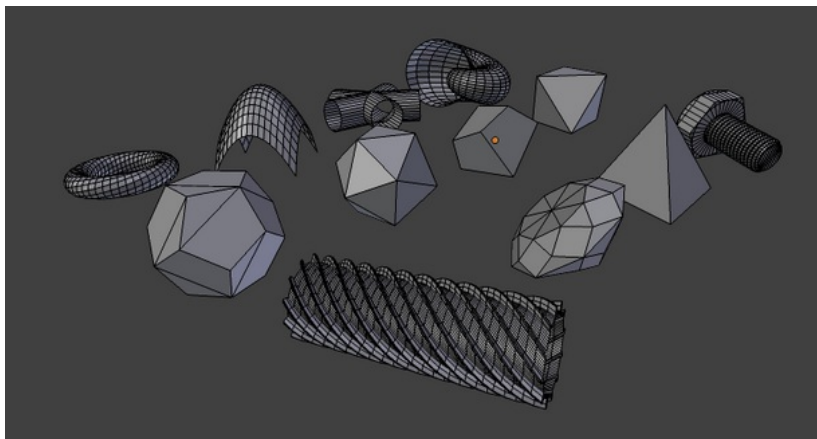
Ruudustik (Grid)

Tavaline ruudustik koosneb n korda m tipust. X-telje ja Y-telje resolutsiooni saab määrata tööriistariulis (Tool Shelf) olevas kontekstipaneelis, mis ilmub pärast objekti loomist; mida kõrgem on resolutsioon, seda rohkem tippe luuakse. Objektid, mida saab ruudustikuga luua, on näiteks maastikud (koos proportsionaalse muutmise tööriista või pinnanihutuse töötlejaga Displace) ja teised orgaanilised pinnad. Ruudustikku saab samuti tekitada, luues tasapinna ning kasutades muutmisrežiimis Edit tükeldamise töötlejat. Aga kasutaja eelistuste (User Preferences) alt võib leida ka maastiku lisa Landscape.

Pärdik (Monkey)

See on kingitus vanalt NaN-ilt Blenderi kogukonnale ning seda peetakse programmeerija naljaks või "üllatusmunaks". See loob pärast nupu Monkey vajutamist pärdikupea. Pärdiku nimi on "Suzanne" ja see on Blenderi maskott. Suzanne on väga kasulik standardne testvõre nagu [Utah' teekann](#) või [Stanfordi küülik](#).

Lisad



Mõned lisadena saada olevad võre primitiivid.

Lisaks geomeetrilistele baasprimitiividele pakub Blender ka pidevalt suurenevat hulka skriptide poolt genereeritud võresid eelinstalleeritud lisadena. Neid saab kasutada pärast sisselülitamist kasutajaeelistuste paneeli User Preferences lisade osast Add-ons (filtreeri võre järgi Add Mesh). Mainime siin vaid mõnda:

Maastik ([Landscape](#))

Lisab maastiku primitiivi. Tööriistariulisse (Tool Shelf) ilmub palju parameetreid ja filtreid.

Toru liigendid ([Pipe Joints](#))

Lisab ühe viiest veetoru liigendi primitiivist. Tööriistariulist saab muuta raadiust, nurka ja teisi parameetreid.

Hammasrattad ([Gears](#))

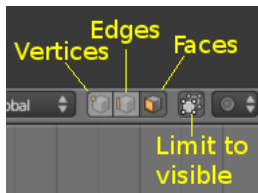
Lisab hammasratta või [tiguülekande](#), mille kuju saab tööriistariulist mitme parameetri kaudu muuta.

Võre komponentide valimine

Valimiseks on mitmeid mooduseid ja saadaolevate tööriistade valik sõltub valiku režiimist (Mesh Select Mode). Kõigepealt tegeleme nende režiimidega ja siis vaatame põhilisi valimise tööriistu.

Valimise režiimid (*Selection Modes*)

Valimise tööriistad vaate päises



Muutmisrežiimis aktiivsed valimise nupud

Muutmisrežiimis on kolm erinevat valimise režiimi. Erinevate valimisrežiimide vahel lülitumiseks saab kasutada vaate tööriistaribal asuvaid nuppe.

On olemas ka miksitud režiim. Selle saamiseks klõpsa ⇧ Shift LMB  soovitud märkimisrežiimi nupul.

Vertices (tipud)

Valitud tippe kuvatakse oranžina, valimata tippe mustana ja viimasena valitud tippu valgena.

Edges (servad)

Selles režiimis tippe värviga ei eristata. Selle asemel on nüüd valitud servad oranžid, valimata servad mustad ja viimasena valitud serv valge.

Faces (küljed)

Selles režiimis on iga külje keskel kuvatud üks valimise punkt. Sellega saab valida külgi. Valitud küljed on oranžid, valimata mustad ja viimati valitu on valge.

Peaaegu kõik muutmise tööriistad on kõigi kolme valimisrežiimi all kättesaadavad. Näiteks saab iga valimisrežiimi puhul kasutada järgmisi teisendusi: pööramine (Rotate), suurendamine (Scale), eendamine (Extrude) jne. Loomulikult üksiku tipu suurendamine ja keeramine miskit nähtavat ega kasulikku ei tee.

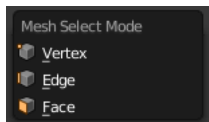
Märkus

Valimisrežiimi nupud on nähtaval ainult muutmisrežiimis olles.

Valimisrežiimi hüpikmenüü

Mode: Muutmisrežiim Edit

Hotkey: Ctrl⇧ Tab



Võre valiku režiimi menüü Mesh Select Mode

Valimisrežiimi saab muuta ka läbi hüpikmenüü.

Select Mode » Vertices (tipud)

Vajuta Ctrl⇧ Tab ja vali hüpikmenüüst Vertices või vajuta klahvi Ctrl⇧ Tab1.

Select Mode » Edges (küljed)

Vajuta Ctrl⇧ Tab ja vali hüpikmenüüst Edges või vajuta klahvi Ctrl⇧ Tab2.

Select Mode » Faces (küljed)

Vajuta Ctrl⇧ Tab ja vali hüpikmenüüst Faces või vajuta klahvi Ctrl⇧ Tab3.

Valimisrežiimide vahel lülitumine

Märgitud komponendid jäävad valimisrežiimide vahetamisel valituks. Aga seda vaid juhul, kui need moodustavad teist tüüpi valimisrežiimis terviku. Näiteks: kui tipu valimise režiimis on valitud kõik ühe külje neli tippu, jääb see külg valituks ka külje valimise režiimi ümber lülitumisel. Kuid kui madalama taseme valimist on mõni jupp puudu, siis kõrgema taseme valimisrežiimi lülitumisel kaob vastav märgistus ära.

Valitud elemendid ja valimisrežiimide vahetamine

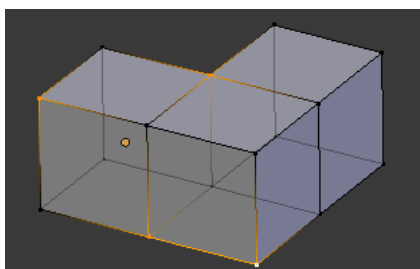
Kui valimisel lülituda lihtsamalt elemendilt keerulisemale, siis jäävad valituks need kõrgema taseme osad, millest madalamal tasemel olid kõik jupid valitud.

Näiteks: kui tipu valimise režiimis on valitud kõik ühe külje neli tippu, jääb see külg valituks ka külje valimise režiimi ümber lülitumisel. Kuid kui madalama taseme valimist on mõni jupp puudu, siis kõrgema taseme valimisrežiimi lülitumisel kaob vastav märgistus ära.

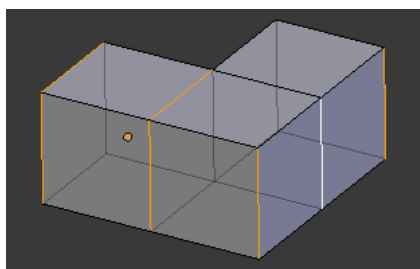
Kuid keerulisemalt lihtsamale lülitudes jäävad alati kõik valitud osad jätkuvalt valituks.

Kui valimisrežiimi nupule vajutamise ajal all hoida Ctrl klahvi, toimub lihtsamalt keerulisemale lülitudes kõigi eelneva valikuga kontaktis olnud elementide valimine.

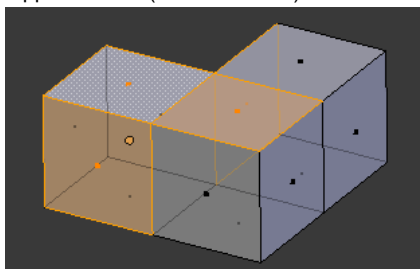
Vaata pilte *Tippuderežiimi (Vertices mode) näide*, *Servadederežiimi (Edges mode) näide*, *Külgederežiimi (Faces mode) näide* ja *Kombineeritud režiimi (Mixed mode) näide*, et näha erinevaid režiime.



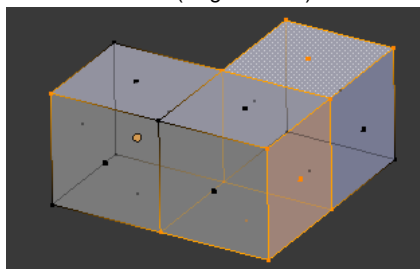
Tippuderežiimi (Vertices mode) näide.



Servadederežiimi (Edges mode) näide.



Külgederežiimi (Faces mode) näide.



Kombineeritud režiimi (Mixed mode) näide.

Lihtne valimine

Mode: Muutmisrežiim Edit

Hotkey: RMB ja Shift RMB

Kõige tavalisem valimise viis on RMB hiireklahviga klõpsamine.

Valikusse lisamine

Valitud elementide hulka uue elemendi lisamiseks hoida all Shift-klahvi ja tee elemendil parempoolse hiireklahviga klõps. Valitud elemendil uuesti klõpsamine muudab selle mittevalituks.

Objektirežiimis on olemas üks unikaalne "aktiivne" element. See on teist värvi ja enamasti viimasena valitud element. Mõningate tööriistade kasutamisel võib see element olla määrava tähtsusega!

Muutmisrežiimis ei saa valida teiste taga olevaid elemente, nii nagu seda sai teha objektirežiimis (kasutades Alt RMB). Muutmisrežiimis olles saab teiste taga olevaid elemente valida siis, kui on sisse lülitatud kuubikukujuline nupp, mis asub valikurežiimi nuppudest paremal pool. See nupp on nähtaval vaid tahke (*solid*), varjutatud (*shaded*) ja tekstuurivaate korral. Ümbritseva kasti (*bounding box*) ja traatvõre (*wireframe*) vaate korral see nupp puudub.

Selle abil saad valida teiste taga olevaid elemente ja vältida kogemata mittenähtaval poolel olevate märgitud elementide teisendamist.

Piirkonna alusel valimine (*Selecting Elements in a Region*)

Mode: Muutmisrežiim Edit

Hotkey: B, BB ja Ctrl LMB - klõpsa ja lohista

Piirkonna järgi valimine laseb objekte valituks teha enda määratud 2D-piirkonnast 3D-vaate suhtes. Valimiala võib olla nii ringikujuline kui ka nelinurk. Vanemates Blenderi versioonides on ringikujuline variant olemas vaid muutmisrežiimis. Uues Blenderis toimivad nii kandiline kui ka ringikujuline märkimisala nii muutmisrežiimis kui ka objektirežiimis.

Märkus

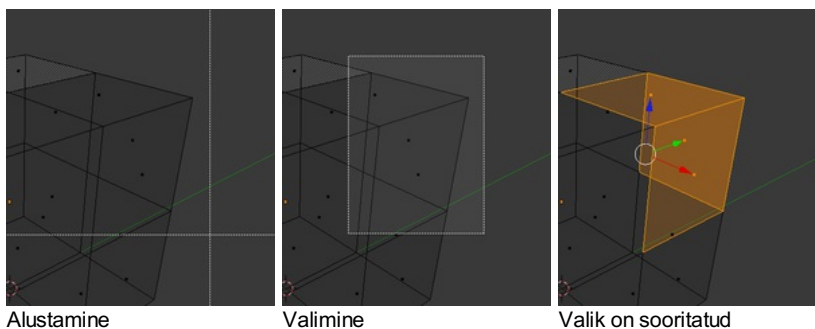
Mõlemi puhul on oluline võimalus valiku piiramine nähtavaga (Limit Selection to visible), mis on 3D-vaate alaosas tahke vaate korral.

Näiteks,

1. olles tahkes vaates ja külje valimise režiimis, valitakse kõik valitud ala sisse jäävad küljed;
2. samas traatvõre varjutuse vaates ja külje valimise režiimis olles valitakse ainult need küljed, mille keskel olev punkt ehk valikusang jääb valitava ala sisse.

Piirdkastiga valimine (*Border select*)

Kandilise alaga valimine (Border Select) on saadaval nii objektirežiimis kui ka muutmisrežiimis. Selle saamiseks vajuta B klahvi. Nüüd lohista hiire vasemat klahvi all hoides üle soovitud ala. Nii muutuvad selle kandilise ala sisse jäävad objektid valituks. Kui enne piirdkastiga valiku tegemist oli aktiivne valikust väljas olev objekt, siis pärast selle valikusse kaasamist on too objekt nüüd nii valitud kui "aktiivne".



Alustamine

Valimine

Valik on sooritatud

Pildil (*Alustamine*) on kastiga valimine (Border Select) aktiveeritud ja näha on ristkujuline kursor. Pildil (*Valimine*) toimub valimine sel teel, et joonistatakse vasemat hiireklahvi all hoides nelinurk. Märgitav ala hõlmab ainult kolme külje valikusangasid. Valiku protseduur jõuab lõpule vasema hiireklahviga klõpsamisel.

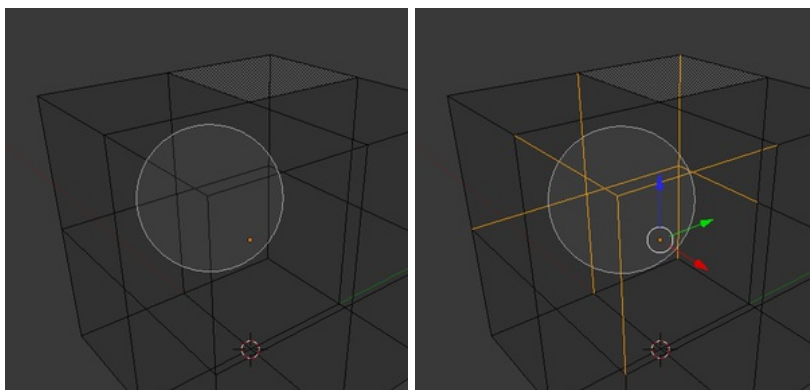
Märkus

Piirdkastiga valimise tööriist lisab uued valitud objektid eelnevale valikule. Nii et ainult uute valimiseks vajuta kõigepealt A klahvi. Objektide valikust välja jätmiseks võid neil klõpsata hiire keskmise klahviga.

Ringikujuline valikuala

Selle aktiveerimiseks on klahv C. Varasemates Blenderi versioonides toimis see ainult muutmisrežiimis, kuid nüüd toimib ka objektirežiimis. Tulemusena näed kursori ümber 2D-ringi. See tööriist toimib sõltumata sellest, mis valimisrežiim parajasti aktiivne on. Vasemat hiireklahvi all hoides ja lohistades valitakse vastava ala sisse jäävad elemendid.

Ringi suurust saad muuta hiire rullikuga kerides või kasutades NumPadi klahve pluss ja miinus.



Enne

Ringikujulise alaga valimine

Pärast

Pildil (*Ringikujulise alaga valimine*) on näide servade valimisest servade valimise režiimis (Edge Select Mode). Nii kui ring puudutab serva, muutub see serv kohe valituks. Valituks muutmiseks peab all hoidma vasemat hiireklahvi.

Mittermärgitaks muutmiseks kasuta keskmist hiireklahvi või kombinatsiooni Alt LMB ja klõpsa ning märgi ala.

Külje valimise režiimis piisab külje valimiseks ringikujulise tööriista abil sellest, kui ring puudutab vaid ühte tippu.

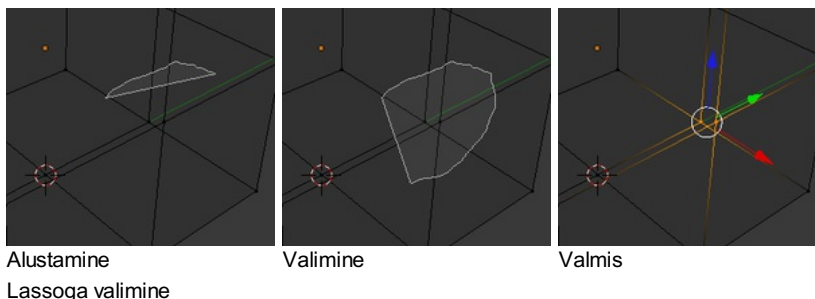
Sellest tööriistast väljumiseks kasuta parempoolset hiire klahvi või vajuta Esc-klahvi.

Vaba käega märgitud ala (*Lasso*)

Vaba käega valimine ehk nn "lasso" erineb ringikujulisest selle poolest, et sellega saad soovitud ala ise joonistada.

Lasso on saadaval nii muutmis- kui ka objektirežiimis. Selle aktiveerimiseks hoia all Ctrl LMB ja märgi soovitud ala. Varasemates Blenderi versioonides tuli sellega valimiseks märgitava ala sisse jätta külje sang ehk keskpunkt, kuid nüüd piisab ka sellest, kui kas või väike osa küljest jääb lasso sisse.

Elemendi valikust välja arvamiseks kasuta kombinatsiooni Ctrl+ Shift LMB.



Alustamine

Valimine

Valmis

Lassoga valimine

Pildil (*Lassoga valimine*) on näide lassoga valimisest tippude valimise režiimis (Vertex Select Mode) olles.

Täiendavad valimise tööriistad

Muutmisrežiimi valimismenüü (Select) sisaldab veel mitmesuguseid tööriistu:

Põhilised tööriistad

Select All/Select None (vali kõik/tühista valik, A)

Vali või tee mittevalituks kõik võre komponendid.

Invert Selection (pööra valikut, CtrlI)

Teeb valitud elemendid mittevalituks ja mittevalitud elemendid valituks.

More (vali rohkem, CtrlNum+)

Suurendab valimit valimis olevate elementide otseses naabruses olevate elementide võrra.

Less (vali vähem, CtrlNum-)

Toimib eelnevaga vastupidiselt. Valim väheneb valimi servast sissepoole.

Erilisemad tööriistad

Mirror (peegel)

Valib ka elemendid, mis on objekti suhtes sümmeetriliselt peegelpildis.

Linked (lingitud, CtrlL)

Valib kõik komponendid, mis on valitute servade abil ühenduses.

Select Random (juhuslik valik)

Valib juhuslikud tipud, servad või küljed etteantud protsendi järgi.

Select Every N Number of Vertices (iga N-s)

Valib iga N-inda tipu eelnevalt valitud tippude hulgast.

Select Sharp Edges (teravad servad)

Valib need servad, mille omavaheline kaldenurk on määratud väiksem. Kaldenurga suurust saab valida pärast selle tööriista valimist tööriistariiuli (*Tool Shelf*) alalt. Mida väiksem on number, seda teravamate nurkadega servad valitakse. **180°** juures toimub kõikide küljega sulgemata servade märkimine.

Linked Flat Faces (sammas tasapinnas, Ctrl⇧ ShiftAltF)

Vali üksteisega ühendatud küljed nendevahelise nurga põhjal. See on kasulik tasapinna moodustavate külgede valimiseks.

Select Non Manifold (topoloogiliselt vigased, Ctrl⇧ ShiftAltM)

Valib tipud, mis pole tervenisti geomeetriaga ümbritsetud. Näiteks raami servad, põhivõrega ühendamata servad ja üksikud tipud. Seda saab kasutada vaid tipu märkimise režiimis.

Interior (sisemus)

Vali küljed, mille kõik servad on ühendatud enam kui kahe küljega.

Side of Active (aktiivse poolel)

Vali kõik elemendid, mis on esimesena valitud elemendiga samal teljel.

Tris (kolmnurgad)

Vali kõik võres olevad kolmnurgad.

Quads (nelinurgad)

Vali kõik võres olevad nelinurgad.

Loose (lahtised)

Vali kõik tipud või servad, mis ei osale külje moodustamises.

Valimine sarnasuse alusel (*Select Similar*)

Mode: Muutmisrežiim Edit

Hotkey: ⇧ ShiftG

Menu: Select » Similar to Selection...

Valib komponendid, millel on esmalt valitute sarnasugused omadused. Pärast tööriista valimist on tööriistariiuli (*Tool Shelf*) alal veel valikute võimalus. Kuvatavad valikud sõltuvad valikurežiimist

Vertex Selection Mode (tipu valimise režiim)

Normal (normaal)

Valib kõik tipud, millel on algselt valitute sarnased normaaliid.

Amount of Vertices in Face (tippude arv küljel)

Valib tipud, mis kuuluvad sarnasuguste naaberkülgede arvuga külgede hulka nagu esmalt valitu. Näiteks kui on valitud tipp, mis ei kuulu ühegi külje hulka, siis tulemuseks valitakse kõik külgede moodustamises mitteosalevad tipud.

Vertex Groups (tipugrupid)

Valib ära kõik tipugrupid, mille koostises algselt valitud tipp või tipud on. Vaata teemat [tipugrupid](#).

Edge Selection Mode (serva valimise režiim)

Length (pikkus)

Valib ära kõik servad, millel on algselt valitutele sama pikkus.

Direction (suund)

Valib ära kõik servad, mis on algselt valitutele samases sihis.

Amount of Vertices in Face (tippude arv küljel)

Valib ära sellised servad, mis on algselt valitutele sama suure arvu külgede koostises. See sarnaneb tippude valimise režiimile, nagu kirjeldatud ülalpool.

Face Angles (külgedevaheline nurk)

Valib kõik küljed, mille omavaheline kaldenurk on sama suur kui nendel, mis juba on valimis.

Crease (viik)

Valib servad, millel on samasugune viigi teravus. Viigi suurust (Crease) kasutab näiteks [pinnatükelduse töötleja \(Subsurf Modifier\)](#).

Seam (õmblus)

Valib kõik servad, millel on sama õmbluse olek nagu nendel, mis on juba valitud. Õmbluse (Seam) kaht sätet kirjeldatakse siin: [UV-tekstuurimine](#).

Sharpness (teravus)

Valib kõik servad, millel on samasugune teravus nagu neil, mis on juba valitud. Teravus (Sharp) on kaheväärtuseline (terav või sujuv) omadus, millest on juttu siin: [Servalõhestaja töötleja \(EdgeSplit Modifier\)](#).

Face Selection Mode (külje valimise režiim)

Material (materjal)

Valib kõik küljed, millel on algselt valitutele sama materjal.

Image (UV-tekstuur)

Valib kõik küljed, mis kasutavad algselt valitutele sama UV-tekstuuri (vaata [UV-tekstuurimise](#) teemat).

Area (pindala)

Valib kõik küljed, millel on algselt valitutele sama pindala.

Perimeter (ümbermõõt)

Valib kõik küljed, millel on algselt valitutele sama ümbermõõt.

Normal (pinnanormaal)

Valib kõik küljed, millel on algselt valitutele samase suunaga normaaliid. Selle abil saab valida samase suunaga küljed.

Co-planar (samas tasapinnas)

Valib küljed, mis on algselt valitutele (peaaegu) samas tasapinnas.

Silmuste valimine (Selecting Loops)

Silmuste märkimiseks on kergelt kasutatavad tööriistad:

Serva- ja tipusilmuse valimine (Edge Loop and Vertex Loop Selection)

Mode: Muutmisrežiim Edit → Tipu- (Vertex) või serva (Edge) valimise režiim

Hotkey: Alt RMB  või CtrlE → Select » Edge Loop

Menu: Select » Edge Loop või Mesh » Edges » Edge Loop

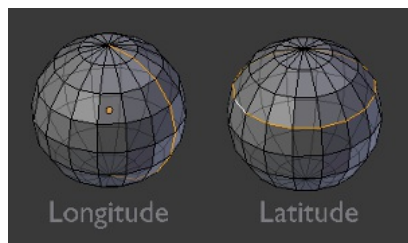
Hiire parempoolse klahviga valikut tehes ja Alt klahvi all hoides muutub valituks selle servaga ühenduses olev servasilmus. Alt+Shift-kombinatsiooni kasutamine hiirega klõpsates lisab uue valiku juba olemasolevasse valimisse.

Servasilmuse saab valida ka menüüst Select » Edge Loop või valides selle hüpikmenüüst Edge Loop Select, mis avaneb klahvikombinatsiooniga (CtrlE).

Tippude valimise režiim

Tippude valimise režiimis saab valimiseks kasutada samasuguseid klahvikombinatsioone nagu varem kirjeldatud servade valimise puhul.

Näide



Piki- ja ristisuunalised servasilmused.

Vasakpoolne kera illustreerib pikisuunalist valikut. Nagu näha, on tegemist avatud silmusega. Nii on see selle tõttu, et poolused katkestavad tsükli ära. Ära katkestavad nad selle seetõttu, et need tipud on ühendatud kahe või siis enam kui kolme küljega. Parempoolsel keral on näha ristisuunaline servasilmus, mis siis ongi n-ö. silmuses. Valitud servi mööda edasi liikudes jõuame lõpuks tagasi samasse punkti.

Küljesilmuse valimine (*Face Loop Selection*)

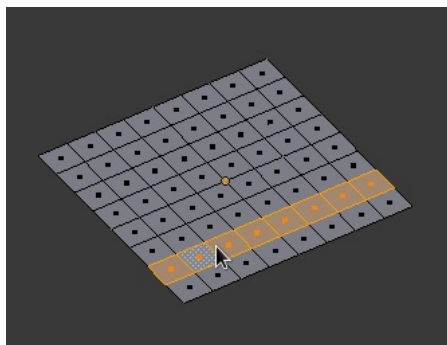
Mode: Muutmisrežiim Edit → Külje (Face) või tipu (Vertex) valimise režiim

Hotkey: Alt RMB

Külgede tsükliks valimiseks tuleb külje valimise režiimis Alt-klahvi all hoides hiire parempoolse klahviga külje "serva" peale klõpsata. Vastavalt servale toimub küljesilmuse suuna valimine.

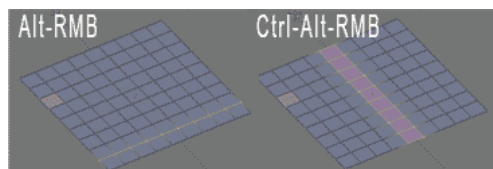
Silmusega külgede valiku saab ära toimetada ka tippude valimise režiimis. Selleks tuleb kasutada kombinatsiooni CtrlAlt + hiire parempoolne klahv.

Näited



Küljesilmuse valik.

Selle küljesilmuse valimiseks kasutati Alt RMB klõpsamist serval ja serva (face) märkimise režiimi. Silmus on valitud serva suunaline.



Alt versus CtrlAlt tippude valimise režiimis.

Küljesilmuse valimiseks saab kasutada ka tippude valimise režiimi. Vaata: (*Alt versus CtrlAlt tippude valimise režiimis*). Sildi "Alt-RMB" juures olevad servad on valitud servasilmusena, vastandina serva ringile. Tippude valimise režiimis teeb vastaspoolse serva servasilmusena valimine valituks terve külgede silmuse.

Tulemusena on valimis siiski näha valitud külgede asemel valitud tippude kogumit, sest valimisrežiim oli tippude valimise oma. Ja nii oleks serva valimise režiimis valimis olnud vastavad servad.

Servarõnga valimine (*Edge Ring Selection*)

Mode: Muutmisrežiim Edit → serva valimise režiim (*Edge select mode*)

Hotkey: CtrlAlt RMB või CtrlE → Select » Edge Ring

Menu: Select » Edge Ring või Mesh » Edges » Edge Ring

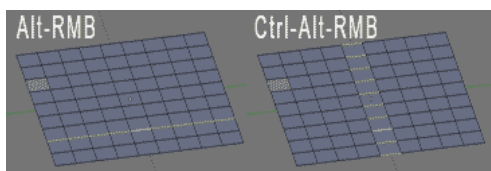
Serva valimise režiimis serva valides ja klahve CtrlAlt all hoides muutub valituks selline tsükiline servade kogum, mis pole omavahel otsapidi ühendatud, vaid on [küljesilmuses](#).

Ka servaringide valimiseks saab kasutada menüüd Select » Edge Ring või hüpikmenüü valikut Edge Ring Select, mis avaneb klahvikombinatsiooni (CtrlE) peale.

Tippude valimise režiim

Tippude valimise režiimis võib kasutada samu klahvikombinatsioone nagu servade puhulgi, aga tulemuseks on vastav küljesilmus.

Näide



Valitud servasilmus (*edge loop*) ja servarõngas (*edge ring*).

Mõlema pildi puhul klõpsati samale servale, aga erinevate käskude tulemusena on saadud valim erinev. Üks põhineb servadel ja teine külgedel.

Silmus piirkonnaks ja vastupidi (*Loop to Region and Region to Loop*)

Mode: Muutmisrežiim Edit → serva valimise režiim (*Edge select mode*)

Hotkey: CtrlE » 9 NumPad ja CtrlE → Select » Loop to Region/Region to Loop

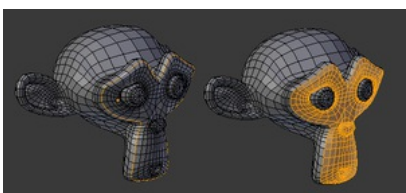
Menu: Select » Loop to Region/Region to Loop või Mesh » Edges » Loop to Region/Region to Loop

Käsk "silmus piirkonnaks" (*Loop to Region*) jaotab võre kaheks piirkonnaks. Iga silmuse korral läheb valimisse väiksem osa jaotatud võrest. Kuigi see asi toimib nii tippude kui ka külgede valimise režiimis, on seal tulemus veider - tuleks jääda servade valimise režiimi...

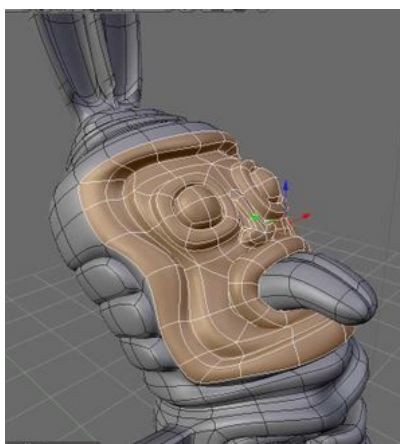
Piirkond silmuseks (*Region to Loop*) on eelnevale täpselt vastupidine asi. See teeb mingi märgitud ala ümber piiri, kus valimis on ainult piiril olevad servad. See funktsioon toimib igasuguse valimisrežiimi korral.

Kogu see värk on selgem, kui illustratsioone vaadata:

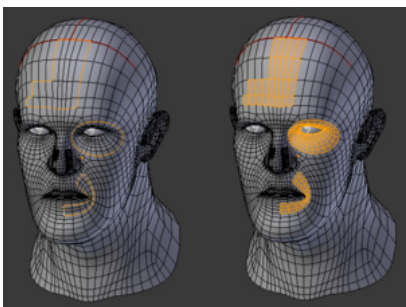
Näide: Silmus piirkonnaks (*Loop to Region*)



Valimine.



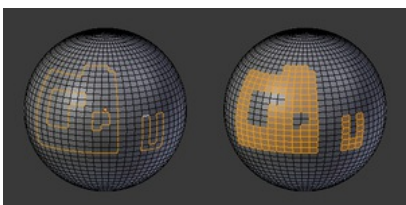
Silmus piirkonnaks.



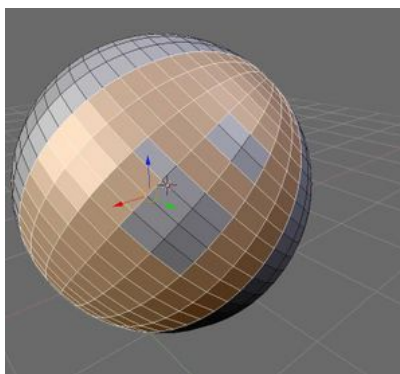
Valik.



Nagu näha, saab see tööriist mitme silmusega kenasti hakkama.

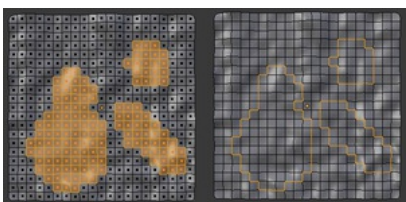


Valik.

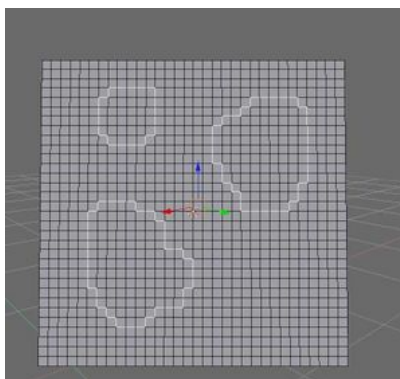


See tööriist saab ka "aukudega" väga hästi hakkama.

Näide: Piirkond silmuseks (*Region to Loop*)



Valik.



See on "tagurpidi keeratud" silmusest-piirkonnaks.

Valitavad elemendid

Nagu oleme näinud [võre struktuuri leheküljel](#), on võred loodud erinevat tüüpi elementidest (kuigi need on kõik omavahel seotud: teatud moel justkui sama alusidee erinevad “vaated”, “esitlusviisid”) nagu “tipud”, “servad” ja “küljed”.

Seega saad valida erinevaid võre osasid kasutades ühte neist kolmest tüübist. Mõista tuleb ühte olulist punkti: *kui sa valid mõne elemendi tüübi (nt mõned servad), valid sa **kaudselt** ka teised vastavad elementide tüübid (nt kõik tipud, mis valitud servi määratlevad, ning ka kõik nende samade servade poolt täielikult määratletud küljed)*. See on väga tähtis, sest mõned servad töötavad ainult tippude, servade ja/või külgedega: kui sa kasutad tippude valiku peal mõnda “külgede” tööriista, siis muudetakse ainult külgi, mida need tipud määratlevad.

Peamiselt valid sa samaaegselt aga ainult ühte tüüpi elemente, mis sõltub sellest, millist “valikurežiimi” sa kasutad. Aga sa saad järjest erinevaid elemente samale valikule lisada, liikudes erinevate valikurežiimide vahel (vaata [altpoolt](#) selle kohta, mis pärast režiimi muutmist valituks jääb), või saad kasutada “kombineeritud” valikurežiimi, mida kirjeldatakse samuti allpool.

Valikurežiimid (Select Modes)

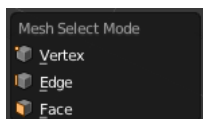
Sul on kaks viisi valikurežiimide vahel liikumiseks:

Valikurežiimi hüpikmenüü

Mode: Muutmisrežiim Edit

Hotkey: Ctrl+ Tab

Muutmisrežiimis Edit on võrede puhul kolm erinevat valikurežiimi – vaata pilti (*Valikurežiimi menüü Select Mode*).



Valikurežiimi menüü
Select Mode.

Select Mode » Vertices (tipud)

Vajuta kombinatsiooni Ctrl+ Tab ja vali hüpikmenüüst tippude sissekanne Vertices või kasuta kombinatsiooni Ctrl+ Tab1 NumPad. Valitud tipud kuvatakse kollastena ning valimata tipud roosadena.

Select Mode » Edges (servad)

Vajuta kombinatsiooni Ctrl+ Tab ja vali hüpikmenüüst servade sissekanne Edges või kasuta kombinatsiooni Ctrl+ Tab2 NumPad.. Selles režiimis tippe ei kuvata. Valitud servad kuvatakse kollastena ning valimata servad mustadena.

Select Mode » Faces (küljed)

Vajuta kombinatsiooni Ctrl+ Tab ja vali hüpikmenüüst külgede sissekanne Faces või kasuta kombinatsiooni Ctrl+ Tab3 NumPad.. Selles režiimis kuvatakse küljed nii, et iga külje keskel oleks valikupunkt, mille abil seda külge saab valida. Valitud küljed kuvatakse kollastena ja valikupunkt on oranž, valimata küljed kuvatakse mustadena.

Kõigis kolmes režiimis saab kasutada peaaegu kõiki muutmise tööriistu. Näiteks saab iga valimisrežiimi puhul kasutada järgmisi teisendusi: pööramine (Rotate), suurendamine (Scale), eendamine (Extrude) jne. Loomulikult üksiku tipu suurendamine ja keeramine miskit nähtavat ega kasulikku ei tee ning seetõttu on mõned tööriistad erinevates režiimides erineva kasuteguriga.

Valikurežiimi päisenupud

Mode: Muutmisrežiim Edit

Panel: 3D-vaate (3D View) päis



Muutmisrežiimi Edit valikurežiimi nupud.

Lisaks saad siseneda erinevatesse režiimidesse valides ühe kolmest tööriistaribas olevast nupust – vaata pilti (*Muutmisrežiimi Edit valikurežiimi nupud*).

Nende nuppude abil saad siseneda ka “**segatud**” ehk “kombineeritud” režiimi, vajutades nuppe kombinatsiooniga ⇧ Shift LMB . Sel moel saad valida samaaegselt tippe, servi ja/või külgi!

Märkus

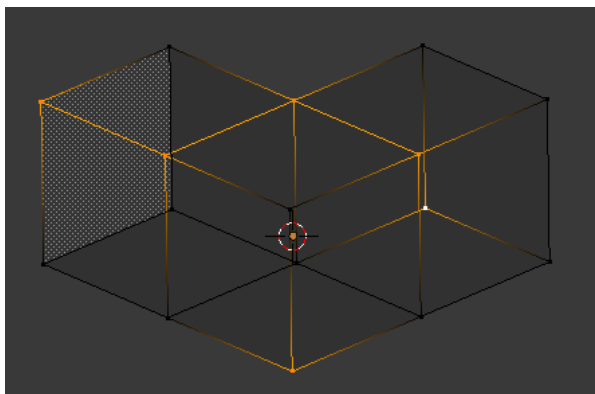
Võrede puhul on “valikurežiimi” nupud nähtavad ainult muutmisrežiimis Edit.

Valitud elemendid pärast valikurežiimi muutmist

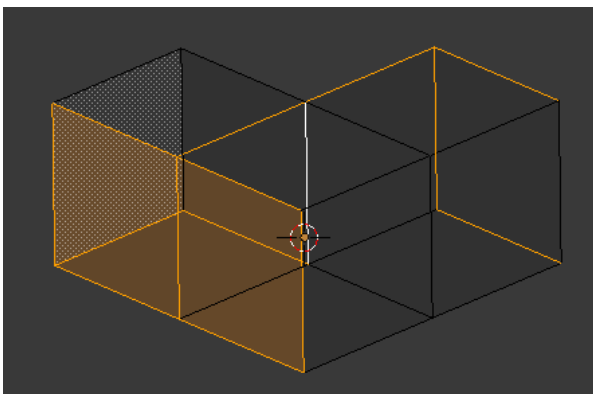
Kui valimisel lülituda lihtsamalt elemendilt keerulisemale (tippudest servadeks ning servadest külgedeks), siis jäävad valituks need kõrgema taseme osad, millest madalamal tasemel olid kõik jupid valitud. Näiteks: kui tipu valimise režiimis on valitud kõik ühe külje neli tippu, jääb see külg valituks ka külje valimise režiimi ümber lülitumisel. Kuid kui pärast madalama taseme valimist on mõni jupp puudu, siis kõrgema taseme valimisrežiimi lülitumisel kaob vastav märgistus ära.

Kuid keerulisemalt lihtsamale lülitudes jäävad alati kõik “keerukama taseme” valitud osad jätkuvalt valituks (näiteks nelinurga neli serva või tippu).

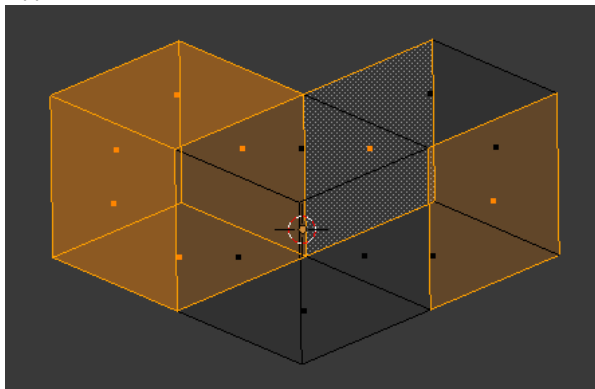
Vaata *Tippuderežiimi näidet*, *Servaderežiimi näidet*, *Külgederežiimi näidet* ja *Kombineeritud režiimi näidet*, et näha erinevaid režiime.



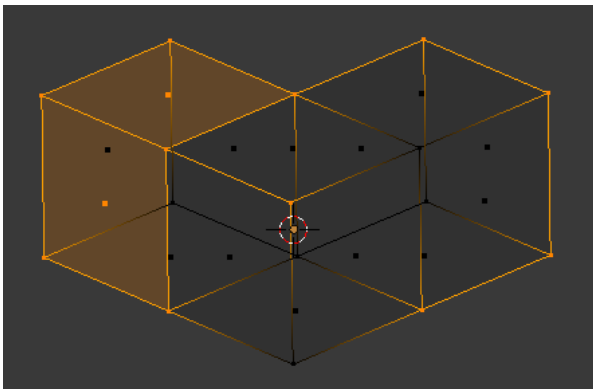
Tippuderežiimi näide.



Servaderežiimi näide.



Külgederežiimi näide.



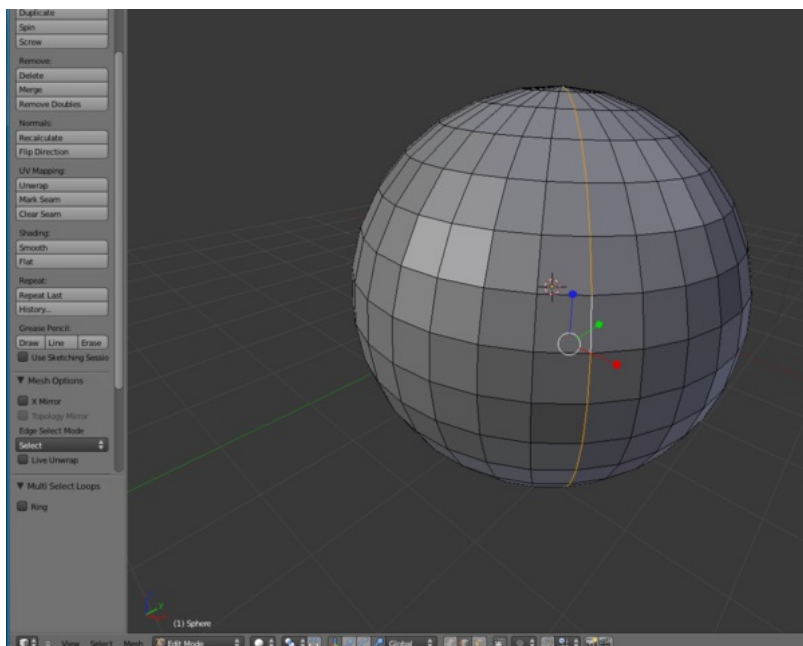
Kombineeritud režiimi näide.

Servade valimine

Servade valimine käib sarnaselt punktide ja külgede valimisega. Selleks tuleb neil parempoolse hiireklahviga klõpsata, olles ise samal ajal serva valimise režiimis.

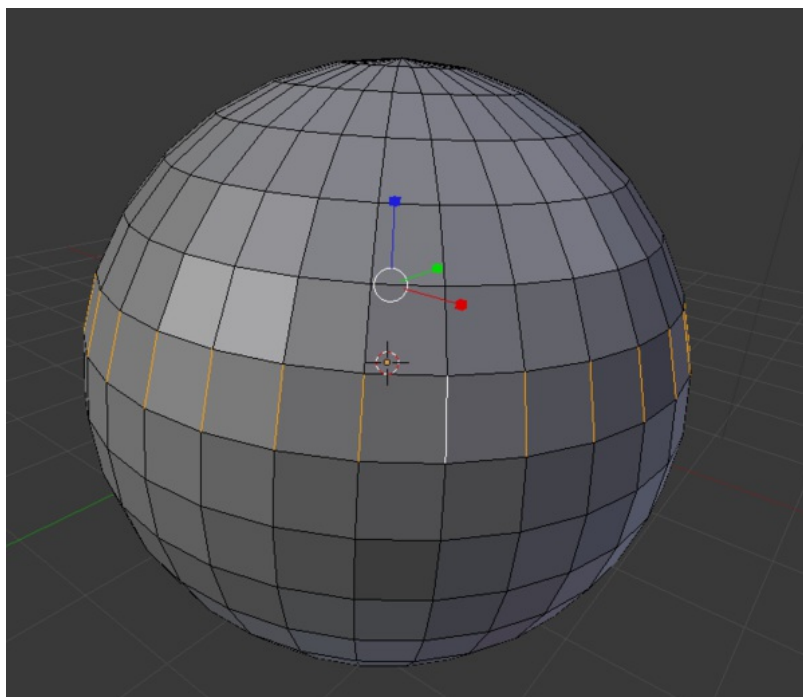
Servasilmused (*Edge Loops*)

Servasilmuse valimiseks tuleb kõigepealt valida üksik serv või kaks tippu ja siis valida menüüst *Select>Edge Loop*



Servarõngad (*Edge Rings*)

Servarõngaste märkimine käib sarnaselt.



Mis juhtuks siis, kui sa tahaksid valikusse võtta kõik küljed, mis on servarõngaste servade vahel? Minnes serva valimise režiimist külje valimise režiimi, tühistatakse valikud. Miks? Nii on see seetõttu, et mitte ühestki küljest polnud valitud kõik neli serva. Iga puhul oli valitud vaid kaks. Nii et külgede valimiseks peaksid kõigepealt valima servarõnga ja siis veel nii ülemise servasilmuse kui ka alumise servasilmuse.

Siiski on olemas lihtsam moodus: pärast servarõnga valimist tuleb minna tippude valimise režiimi. Nüüd juhtub see, et iga valikus olev serv tõlgendatakse kaheks valikus olevaks tipuks. Seis on selline, et valitud on külgede kõik neli tippu: Blenderil puudub info selle kohta, millised servad olid valitud ja millised mitte. Tulemuseks on see, et minnes tippude valimise režiimist külgede valimise režiimi, on kõik küljed valitud, kuna valikus olid kõik nende külgede tipud.

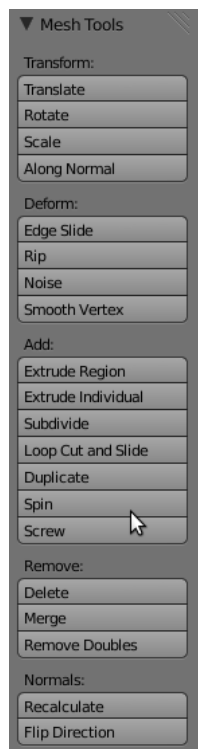
Võre muutmine

Blender pakub võre muutmiseks mitut sorti tööriistu. Need asuvad võre tööriistade Mesh Tools paletis, võre menüüs (*Mesh*) 3D-vaate päises, 3D-vaate kontekstimenüüdes ning neid saab esile kutsuda ka klaviatuuri kiirvalikutega.

Pea meeles, et “teisendamise täpsuse/nakkumise” nupud (Ctrl ja/või ⇧ Shift) töötavad ka kõigi nende keerulisemate võimaluste puhul. Kuid enamikul neist puudub [telgede lukustamise](#) võimalus ning mõned ei võta ka arvesse [objekti keskpunkti](#) ja/või [teisendussuunda](#).

Need teisenduse tööriistad asuvad menüüriba võre (Mesh) menüü teisenduse osas Transform. Pane tähele, et mõningaid neist saab kasutada ka teiste muudetavate objektide nagu kõverate, pindade ja sõrestike peal.

Tööriistade tüübid



Võre tööriistad
(Mesh Tools)

Võre tööriistad asuvad erinevates kohtades ning neid saab kasutada ka klaviatuuri kiirvalikute abil.

[Lisamise ja jagamise tööriistad:](#)

[Teisendamise ja moonutamise tööriistad:](#)

- Liiguta (*Translate*)
- Pööra (*Rotate*)
- Muuda suurust (*Scale*)
- Peegelda (*Mirror*)
- Suru kokku/lamenda/mööda normaali (*Shrink/Flatten/Along Normal*)
- Lükka/tõmba (*Push/Pull*)
- Keraks (*To Sphere*)
- Aja kiiva (*Shear*)
- Vääna (*Warp*)
- Nihuta serva (*Edge Slide*)
- Mära (*Noise*)
- Sujuv tipp (*Smooth Vertex*)
- Serv tagurpidi (*Edge Flip*)
- Pööra serva (*Rotate Edge*)
- Tekita serv/külg (*Make Edge/Face*)
- Täida (*Fill*)
- Täida ilusti (*Beauty Fill*)
- Tahkesta (*Solidify*)
- Nelinurgad kolmnurkadeks (*Quads to Tris*)
- Eenda ala (*Extrude Region*)
- Eenda üksik (*Extrude Individual*)
- Tükelda (*Subdivide*)
- Silmuse lisamine/nihutamine (*Loop Cut/Slide*)
- Nuga (*Knife tool*)
- Duplitseeri (*Duplicate*)
- Vurr (*Spin*)
- Krui (*Screw*)

[Liitmise ja eemaldamise tööriistad:](#)

- Kustuta (*Delete*)
- Ühenda (*Merge*)
- Eemalda topeltelemendid (*Remove Doubles*)
- Kolmnurgad nelinurkadeks (*Tris to Quads*)

[Lahutamise tööriistad:](#)

- Rebi (*Rip*)
- Lõhesta (*Split*)
- Lahuta (*Separate*)

Võre tööriistadele ligi pääsemine

Võre tööriistade palett

Kui sa valid võre ja lähed klahviga ⇌ Tab muutmisrežiimi, muutub tööriistariiul Tool Shelf objekti tööriistadest Object Tools võre tööriistadeks Mesh Tools. Need on vaid mõned võre muutmise tööriistadest.

Menüüd

Võre menüü Mesh asub päise ribas. Mõnedesse menüüdesse pääseb ka kiirvalikutega:

CtrlF näitab külje (*Face*) menüüd

CtrlE näitab serva (*Edge*) menüüd

CtrlV näitab tipu (*Vertex*) menüüd

Teisendamise ja moonutamise tööriistad

Need tööriistad liigutavad olemasolevaid võre osasid erinevatel viisidel.

Teisendamise tööriistad

Liiguta, pööra, muuda suurust

Valitud komponentide liigutamiseks, pööramiseks ja suuruse muutmiseks kasuta nuppe Translate (liiguta), Rotate (pööra) ja Scale (muuda suurust) [teisendusmanipulaatoreid](#) või kiirvalikuid:


vastavalt G, R ja S.

Pärast seda muudetakse tööriistariiuli (*Tool shelf*) alumine osa tööriistaspetsiifiliseks paneeliks (st näidatakse suuruse muutmise (Resize) paneeli, kui oled komponentide suurust muutnud). Seda paneeli saad kasutada muutuste täpsustamiseks, muutuse piiramiseks mõnede telgedega, proportsionaalse muutmise sisse ja välja lülitamiseks jne.

Peegeldamine (*Mirror*)

Geomeetria peegeldamiseks (Mirror) kasuta kiirvalikut CtrlM.



Pärast seda, kui see tööriist muutub aktiivseks, vali peegeldamise telg, sisestades X, Y või Z.

Saad geomeetria ka interaktiivselt peegeldada, hoides all MMB  ning lohistades hiirt soovitud peegeldamise suunas.

Serva nihutamine (*Edge Slide*)

Serva nihutamine (Edge Slide, CtrlE » Edge Slide)

nihutab ühe või enama serva üle naabruses olevate külgede vastavalt mõnele piirangule servade valimise osas (st valik peab olema loogiline, loe altpoolt).

LMB  kinnitab nihutuse ning RMB  või Esc tühistab.

Sellel tööriistal on ka numbriline väärtus, mida kuvatakse 3D-vaate jaluses ja tööriistariiulis (pärast kinnitamist). Täpsema tulemuse saamiseks võib sisestada numbrilise väärtuse käsitsi, vahemikus -1 kuni 1.

Suru kokku/lamenda (*Shrink/Flatten*)

Valitud tippe saab mööda nende normaale kokku suruda ja lamendada, kasutades nuppu Along Normals või kiirvalikut AltS.

Serva pööramine (*Edge Flip*)

Serva pööramine (Edge Flip, ⇧ ShiftCtrlF)

Vali kaks kõrvuti olevat kolmnurka ning pööra siis ühine serv vastasnurka

Pööra serva (Rotate Edge, CtrlE >> rotate edge CW/CCW (pööra päripäeva/vastupäeva))

Kui valitud on serv või kaks kõrvuti olevat külge, kasuta seda tööriista, et pöörata servade grupis jagatavat serva.

Geomeetria moonutamine

Suru kokku/lamenda (*Shrink/Flatten*)

Valitud tippe saab mööda nende normaale kokku suruda ja lamendada kasutades nuppu Along Normals või kiirvalikut AltS.

Lükka/tõmba (*Push/Pull*)

Sarnaselt valikuga 'suru kokku/lamenda' tähistab see teisendus kõigi valitud elementide liigutamist mööda sirget, mis ühendab nende algset asukohta nende asukohtade keskmisega. Kõiki liigutatakse sama väärtuse võrra ning teisendamist viiakse läbi hiirega lohistades. Tulemuseks on miski, mis meenutab natuke suuruse muutmise tööriista efekti, kuid veidi suurema moonutusega.

Pea meeles, et erinevalt eelnevatest teisendustest saad sa selle teisenduse mõne teljega [lukustada](#) – kuigi see lukustamine ei anna erilist tulemust (erandina võib-olla "tasapinnalise lukustamise" korral).

Vääna (*Warp*)

Hotkey: ⇧ ShiftW

Menu: Mesh/Curve/Surface » Transform » Warp

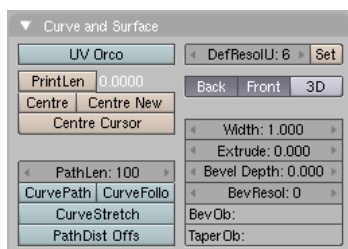
Väänamise teisendus Warp on kasulik väga erilistel juhtudel. See vääna valitud elemendid ümber 3D-kursori (ja alati 3D-kursori, võtmata arvesse teisenduse keskpunkti sätet). Samuti sõltub see vaatest. Punktid, mis joonduvad kursoriga vertikaalselt, jäävad omaale kohale. Iga punkti kaugus kursori **horisontaalsest asukohast** vastab sellele raadiusele, kui kaugel punkt asub kursorist pärast tööriista sisselülitamist. Seejärel vääna tööriist punkti ümber kursori. Väärtus 360 vääna võre täisringina, nii et kõige kaugemal vasakul ja paremal olevad punktid joonduvad omavahel ja kursori asukohaga.

Tööriista kasutamiseks aseta kursor vaates kohta, kus peaks asuma väände keskkohat. Lülita tööriist sisse ning liiguta seejärel kursorit või sisesta nurga väärtus, mille võrra võret tuleks väänata.

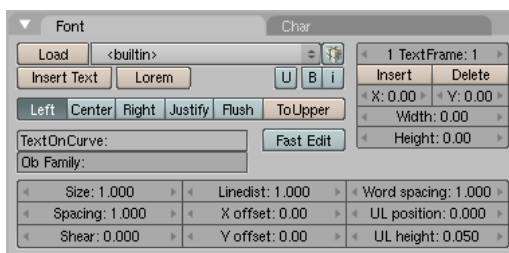
Näide

Ringikujuliselt mähitud tekst on kasulik, kui soovid luua lendavaid logosid, kuid ilma väänamise tööriistata oleks selle modelleerimine keeruline. Näitena vääname lause *Amazingly Warped Text* ("Imeliselt väänatud tekst") ümber kera.

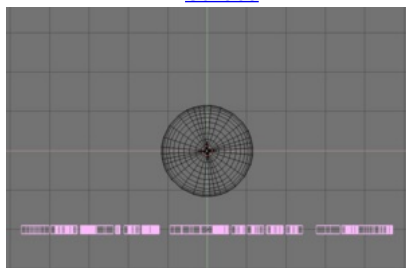
- Lisa kõigepealt kera.
- Seejärel lisa eestvaates tekst ning määra muutmise kontekstis Editing kõvera ja pinna paneelis [Curve and Surface](#) eendamise Extrude väärtuseks 0.1 – tekst muutub kolmemõõtmeliseks. Servale ilusa kandi saamiseks määra kantimise sügavuse Bevel Depth väärtuseks 0.01. Kandi resolutsiooni Bev Resol väärtuseks määra 1 või 2, et kant oleks sujuv, ning vähenda resolutsiooni, et tippude arv hiljem objekti tükeldamisel liiga suureks ei läheks (tükeldamiseks kasuta *Kõvera ja pinna paneeli Curve and Surface ja Fondi paneeli Font*).
- Konverteeri objekt kõverateks ja seejärel võreks (kaks korda AltC), sest väänamine ei tööta teksti ega kõverate peal.
- Tükelda võret kaks korda (W » Subdivide Multi » 2), et geomeetria puhtalt ja ilma artefaktideta kuju muudaks.



Kõvera ja pinna paneel [Curve and Surface](#).

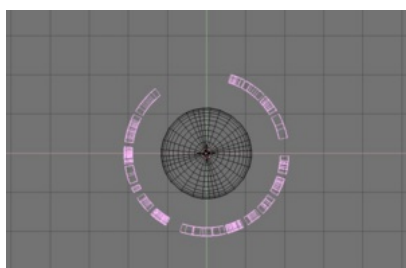


[Fondi paneel](#).



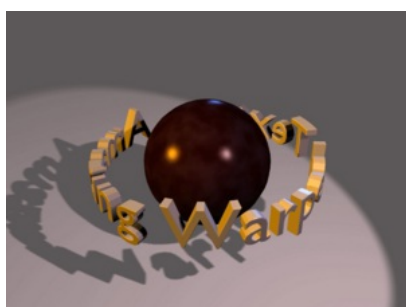
Teksti ja kera vaade ülalt.

- Mine ülaltvaatesse ja liiguta võre 3D-kursorist eemale. Kaugus määrab väände raadiuse. Vaata pilti (*Teksti ja kera vaade ülalt*).



Väänatud tekst.

- Mine võrega muutmisrežiimi Edit (⇐ Tab) ja vali klahviga A kõik tipud. Vajuta ⇧ ShiftW väänamise tööriista sisselülitamiseks. Liiguta hiirt vasakule või paremale, et interaktiivselt väände määra muuta; pilt *Väänatud tekst*.



Lõplik renderdus.



Nüüd võid minna kaamera vaatesse, lisada materjalid ja valgused ning renderdada - pilt *Lõplik renderdus*.

Aja kiiva (Shear)

Hotkey: CtrlAlt⇧ ShiftS

Menu: Object/Mesh/Curve/Surface » Transform » Shear

Kiivaajamise teisendus Shear ajab valitud elemendid kiiva (muutmisrežiimis Edit, mõjutab tippu/servi/kontrollpunkte). Nagu teisedki teisenduse tööriistad, kasutab see vaate koordinaadistikku ning selle keskkohas asub keskpunktis: kiiva ajamine toimub keskpunkti läbiva vaate X-telje suunas. Kõik, mis on sellest teljest "ülevalpool" (st selle Y-telje asukoht on positiivne), liigub (läheb kiiva) samas suunas nagu su hiire kursor (aga alati X-teljega paralleelselt). Ja kõik, mis on sellest X-teljest "allpool", liigub vastassuunas. Mida kaugemal X-teljest element on, seda rohkem see liigub.

Kui tööriist muutub aktiivseks, liiguta hiirt paremale või vasakule, et kiivamineku kallet määrata. Kui soovid, et efekt töötaks horisontaalse asemel vertikaalsel teljel, vajuta klahvi MMB  ning liiguta seejärel hiirt üles või alla. Teise võimalusena võid sisestada numbrilise väärtuse 0 kuni lõpmatus. Tööriista kinnitamiseks vajuta LMB .

Kerakujuliseks (*To Sphere*)

Mode: Muutmisrežiim Edit

Panel: Võre tööriistad Mesh Tools (muutmise kontekst Editing, F9)

Hotkey: ⇧ ShiftAltS

Menu: Mesh/Curve/Surface » Transform » To Sphere

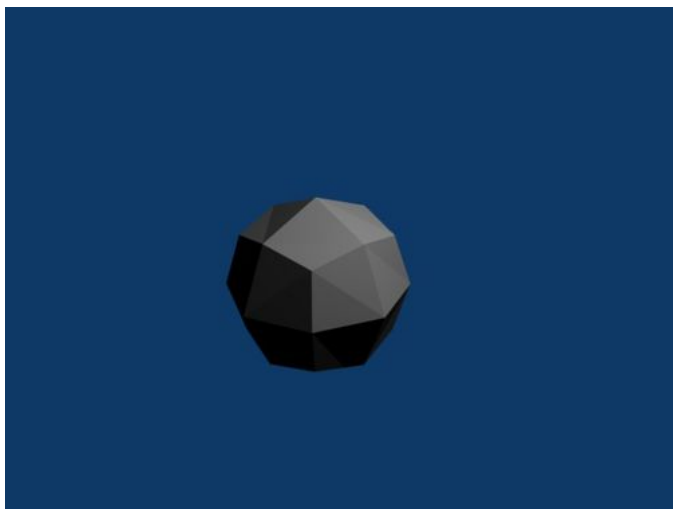
See käsk "kerastab" valitud võre elemendid. Kõigepealt leitakse kõigi elementide keskmine asukoht ning seejärel liigutatakse neid kõiki nii, et nende keskmine kaugus leitud keskpunktist oleks sama. Kasutades väärtust 1, asetatakse kõik tipud sellest keskpunktist võrdsele kaugusele ja tekib kerajas kuju.

Kui tööriist muutub aktiivseks, lohista hiirt vasakule või paremale, et mõju interaktiivselt määrata, või sisesta käsitsi määramiseks väärtus nulli ja ühe vahel.

Näide

Selle tööriistaga saad tükeldatud kuupidest luua kerasid. Alusta kõigepealt [kuubiga](#). Alustame tühjalt lehelt ja kustutame selleks esmalt kõik objektid (CtrlX) ning lisame seejärel kuubi.

- Vajuta muutmisrežiimi Edit sisenemiseks klahvi ⇧ Tab.
- Ole kindel, et kõik kuubi tipud on valitud, vajutades kaks korda järjest klahvi A. Seejärel mine muutmise konteksti Editing, vajutades F9. Nüüd peaksid nägema võre tööriistade paneeli Mesh Tools.
- Tükelda kuup vajutades võre tööriistade paneelis Mesh Tools olevat tükeldamise nuppu Subdivide või kiirvalikuga W » Subdivide. Võid teha seda nii mitu korda kui soovid - mida rohkem sa tükeldad, seda sujuvam kera on.
- Nüüd vajuta Ctrl⇧ ShiftS ja liiguta hiirt vasakule või paremale, et interaktiivselt "kerastamise" mõju muuta (või kirjuta väärtus otse, näiteks "1.000", et saada sama tulemus nagu allpool) – soovitatavalt kasutades keskpunktina mediaanpunkti Median Point!
- Teise võimalusena võid kasutada keraks tegemise nuppu To Sphere (võre tööriistade paneelis Mesh Tools). Kera tegemiseks vali väärtuseks "100". Pea meeles, et sa ei tohiks liigutada oma 3D-kursorit – sellisel juhul ei ole tulemuseks mitte kera, vaid kera osa.



Lõplik madalaresolutsiooniline kera!

Müra (*Noise*)

Müra (Noise) kasutab materjali esimeses tekstuuri lahtris olevat tekstuuri sisendina, mille alusel moonutada valitud tippe.

Et see tööriist töötaks, peab võrele olema määratud materjal ja tekstuur. Et tekstuur ei mõjutaks materjali omadusi, võid selle tekstuuri menüüst välja lülitada. Moonutust saab juhtida, muutes tekstuurilaotust paneeli Mapping kaudu, või tekstuuri enda seadeid ja tüüpi (nt pilved, marmor jne).

Märkus

[Nihketöötleja](#) on müra tööriista mittedestruktiivne alternatiiv.

Uuri ka ANT Landscape [lisa](#).

Silu (*Smooth*)

Silu tippe (Smooth Vertex, W » Shade Smooth või ⇧): silub valitud tipud, lamendades tippude nurgad.

Pärast kinnitamist ilmuvad tööriistariiulisse alternatiivsed võimalused, mille abil saad mõju piirata kindla teljega ning määrata silumisiteratsioonide arvu.

Märkus

[Silumistöötleja](#), mida saab piirata ühe tipugrupiga Vertex Group, on silumistööriista mittedestruktiivne alternatiiv.

Kustutamine ja ühendamine

Nende riistadega saab objekti osi kustutada ja üheks kokku liita.

Kustutamine

Kustutamise käsk (Delete, X või Del)

Kustutab valitud tipud, servad või küljed.

Sellele toimingule võib ka piiranguid seada:

All (kõik)

Laiendab toimingut kogu võrele. Objekti ennast siiski ei kustutata; nii saab objekti loomist nullist alustada, nii et jäävad siiski alles mõningad omadused nagu materjalid, töötlejad jms.

Selle abil saab olemasolevad vajalikud omadused kiiresti uude loodavasse objekti "eksportida". Tee lihtsalt olemasolevast võreobjektist koopias, näiteks kõverale allutatud sinisest kuubikust, millel on ka mõned töötlejad; muuda režiim klahvi ⇨ Tab abil muutmisrežiimiks, kustuta kõik tipud ja lisa näiteks kera, et kasutada kõiki kuubiku muid endisi omadusi.

Edges & Faces (servad ja küljed)

Piira toimingut ainult valitud servadele ja nendega külgnevatele külgedele.

Only Faces (ainult küljed)

Kustutab ainult valitud küljed.

Edge Loop (servasilmus)

Kustutab kogu servasilmuse. Kui valim pole servasilmus, ei juhtu seda tehes mitte midagi.

Liitmine

Liitmine (Merge, AltM)

Ühendab valitud tipud üheks tipuks.

Hüpikmenüüs küsitakse, millisesse keskpunkti paigutada ühendatud tipp. Hiirega klõpsamise asemel võid kiiremaks toimetamiseks ka vastavat numbriklahvi vajutada. Näiteks valik esimene (First, 1), viimane (Last, 2) või keskkoh (Center, 3).

Duplikaatide eemaldamine (Remove Doubles) (W » Remove doubles või 4)

eemaldab kõik valikus üksteisega kattuvad tipud.

Valitud tipud ei pea ühel joonel asuma. Pärast selle toimingu tegemist ilmub tööriistakasti (Tool Shelf) parameeter Merge threshold (liitmise lävi). Lävi määrab ära selle, kui suur võib maksimaalselt olla tippude vaheline vahemaa, et nad veel kokku liidetaks.

Kolmnurkade nelinurkadeks muutmine

Tris to Quads AltJ See liidab teineteisega külgnevad kolmetipulised küljed neljatipulisteks. Seda tööriista saab kasutada terve hulga valimis olevate kolmnurkade nelinurkadeks ühendamiseks.

Võib valida ka ainult kaks teineteisega külgnevat kolmnurka ja siis muuta need üheks nelinurgaks klahvikombinatsiooniga AltJ.

Lisamine ja eemaldamine

Nende tööriistade abil saab võrele keerukust lisada.

Uute servade ja külgede loomine

Loo uus serv/külg (Make Edge/Face, F)

Selle tööriistaga saab võrele geomeetria lisada, kui on valitud sobivad küljed, servad või tipud järgmiste reeglite järgi:

- 2 punkti moodustavad serva
- 3 punkti moodustavad kolmnurga
- 4 punkti moodustavad külje
- 2 serva moodustavad külje

Täida (Fill, AltF)

Täidab valitud servadega piiratud ala uute külgedega. Pane tähele, et uued küljed luuakse trianguleerimise põhimõttel ja on seega kolmnurksed.

Täida ilusti (Beauty Fill, ⇧ ShiftAltF)

Korrastab valitud kolme- ja neljatipuliste külgede sisemist servade paigutust. Seda tööriista võib kasutada mitu korda järjest, kuni paremat jaotust enam ei leita.

Eenda (*Extrude*)

Eenda ala (Extrude Region, E või AltE » Region või 1)

Eendab valitud tippudest uue vormi, mille suund on sama valitud tippude keskmistatud normaali.

Märkus

Valikut eendatakse kas (1) mööda valitud külgede keskmistatud normaali suunda või (2) kui valik ei piiritle tervet külge, siis mööda ühe serva normaali. Muudel juhtudel saab eendamise piirata vaid ühe telje suunaga (näiteks X ainult X-telje suunaliseks piiramiseks või ⇧ ShiftX X-teljega risti oleva YZ-tasandiga piiramiseks).

Kui on soov küljest eendamist teha globaalse Z-telje suunal, tuleb Z klahvi kaks korda vajutada. Esimene vajutus lülitab välja külje enda Z-telje piirangu ja teine lülitab sisse globaalse koordinaadistiku Z-telje piirangu.

Eenda üksikult (Extrude Individual, AltE » Individual Faces või 2)

Eendab algetest külgedest uued küljed piki iga külje individuaalset normaali.

Tahkesta (*Solidify*)

Tahkesta (Solidify)

See eendab valitud küljed, nii et neile tekib justkui "paksus". See funktsioon on saadaval ka mittedestruktiivse töötlejana (*modifier*). Pärast selle tööriista kasutamist saab määrata paksendamise väärtuse numbriliselt tööriistapaletilt.

Tükelda (*Subdivide*)

Tükelda (Subdivide, W » Subdivide või 1)

Tükeldab valitud küljed.

Lõigete arvu ja sujuvuse saab valida tööriistapaletilt (Tool Shelf).

Fraktaalne jaotus (Fractal) - seda saab kasutada түкeldamisele juhuslikkuse lisamiseks.

Erinevad nurga lõike mustrite (Corner Cut Patterns) valikud (Fan, Inner Vertex ja Path) annavad keerulisemate valimite puhul erinevaid tulemusi.

Silmuse lisamine (*Loop Cut*)


Silmuse lisamine ja libistamine (Loop Cut and Slide, CtrlR)

Lisab servatsükli ehk silmuse ja libistab seda mööda valitud servi.

Sellel tööriistal on olemas libistamise väärtus, mida kuvatakse 3D-vaate akna jaluses ja peale muutuse kinnitamist tööriistariiulil Tool Shelf. Täpsema tulemuse saamiseks võib sisestada numbrilise väärtuse vahemikus -1 kuni 1. Seda saab teha enne libistamise kinnitamist klaviatuuril trükkides. Pane tähele, et koma asemel tuleb kasutada punkti. Esc-klahvi vajutamine jätab uue tekkinud tsükli täpselt tsükli aluseks olnud servade keskkoha.

Nuga (*Knife Tool*)

Nuga (Knife Tool, K ja ⇧ ShiftK)

Lohista kursorit vasema hiire klahviga, hoides all K klahvi. Tulemusena tekib servadesse lõikekoht sinna, kust hiirega üle lohistati. Pärast LMB  vabastamist ilmuvad tööriistapaletile järgmised valikud:

Tüüp (Type):

- Täpne (*Exact*) - teeb ühe lõike täpselt sealt, kus kursor määrab serva lõikumiskoha
- Keskkohast (*Midpoints*) - teeb kursoriga lõike mööda tõmmatud joonega lõikuvate servade keskkohati
- Mitmekordselt (*Multicut*) - lõikab serva väiksemateks võrdseteks segmentideks määratud arvu kordi.

Duplitseeri (*Duplicate*)

Duplitseeri (Duplicate, ⇧ ShiftD)

Teeb märgitud tippudest koopiad ja laseb siis neid nihutada. Koopia jätmiseks originaaliga samasse kohta vajuta Esc klahvi.

Tööriistapaneeli (Tool Shelf) alal on valikud vektori nihke (Vector *offset*), proportsionaalse muutmise (Proportional Editing), duplitseerimise viisi (Duplication Mode) (mittetoimiv?) ja teljepiirangute (Axis Constraints) jaoks.

Neljatipuliste külgede kolmetipulisteks muutmine

Nelinurgad kolmnurkadeks (Quads to Tris, CtrlT)

Teeb valitud neljatipulised küljed kolmetipulisteks nii, et ühest neljatipulisest saab kaks kolmetipulist.

Vurr (*Spin*)

Vurr (Spin, AltR)

Pöörab valitud tippe ümber 3D-kursori. Toimingu tulemus sõltub 3D-vaate orientatsioonist.

Astmete (Steps) ja kraadide (Degrees) arvu, samuti keskkoha (Center) asukohta ja telje (Axis) suunda saab määrata tööriistapaneelilt (Tool Shelf).

Kruvi (*Screw*)

Kruvi (Screw)

Kruvib märgitud asju ümber 3D-kursori. Toimingu tulemus sõltub 3D-vaate orientatsioonist. Valitud peab olema rida servadega ühendatud tippe, kuid need ei tohi moodustada silmust.

Märkus

Kruvi (*Screw*) tööriist Blender 2.5-s on varasemate versioonide sama tööriistaga võrreldes erinev selle poolest, et suuna määramiseks pole tarvis määrata serva.

Geomeetria eraldamine (*Separating Geometry*)

Nende tööriistadega saab objekti küljest tükke eraldada, nii et need muutuvad uueks eraldi objektiks.

Eraldamine (*Separate*)

Separate (eralda, P)

Eralda (Separate) käsk eraldab märgitud komponendid ja teeb neist uue objekti. Võimalik on määrata, kuidas valitud elemendid eraldatakse.

Selected (valitud)

Eraldab märgitud osad.

By Material (materjali alusel)

Kui ühes võres on kasutusel mitu materjali, siis saab igast erinevat materjali kasutavast osast eraldi objekti.

All Loose Parts (kõik eraldi osad)

Kui ühe võre sees on mitu omavahel ühendamata punktide kogumit, siis saab need eraldi objektideks teha.

Poolitamine (*Split*)

Split (poolita, Y)

Poolitamine (Split) ühendab märgitud punktid põhivõrest lahti ilma eraldi objekti tegemata. Kustutatakse kõik küljed, mis ühendavad valitud tippe valimata tippudega.

Lõhestamine (*Rip*)

Rip (lõhesta, V)

See tööriist töötab ainult siis, kui on ära märgitud ainult üks serv või ainult üks servasilmus (*Edge Loop*). Sellisel juhul tekitatakse võresse märgitud serva(de) koha peale lõhe, nii et algse valitud serva asemel on meil kaks serva kummalgi pool lõhet.

Märkus

Nii serva libistamise (Edge Slide) kui ka lõhestamise (Rip) tööriista jaoks peab olema tehtud mõistlik objekti osade valik. Valim ei tohiks sisaldada pooluseid (tippe, millel on kolm, viis või rohkem serva), ristuvaid või paralleelseid silmuseid ega enesega ristumisi (Non Manifolds). Kõige õigem viis ühe korraga õigete servade märkimiseks on kasutada mõnda varianti, mis on kirjeldatud [võre valimise tööriistade](#) peatükis (näiteks ⇧ ShiftAltE.)

Topoloogia muutmine (*Retopologizing*)

Märkus

Blenderi versioonis 2.5 on topoloogia muutmise tööriist asendatud paremate võrega nakkumise võimalustega. See lehekülg muutub vastavalt sellele, kuidas topoloogia muutmise tööriistu uuemates Blenderi versioonides täiendatakse.

Topoloogia muutmine on modelleerimise suhteliselt tavaline osa. Sageli luuakse mudel, mille rõhk on vormil ja detailidel, kuid samas pole selle topoloogia ehk servade voolavus ideaalne või on võre väga tihe ja seega ebapraktiline. Modelleerijad saavad luua uue madalama resolutsiooniga võre, mis vastab originaalvõre vormile.

Võrega nakkumine (*Mesh Snapping*)

Nakkumise sisselülitamisel ning nakkumise sihtmärgi seadmisel küljeks (*Face*) projitseeritakse võre tipud vaateaknas (vaate Z-teljel) lähimale pinnale.

See võimaldab sul vabalt modelleerida, ilma et peaksid vormi pärast muretsema, ja keskenduda topoloogia.

Vaata [nakkumise osa](#)

Sobitamise töötleja (*Shrinkwrap Modifier*)

[Sobitamise töötleja](#) on käsikäes külgedega nakkumisega väga kasulik. Kui uuele võrele on tehtud muudatusi, nii et nakkumine on välja lülitatud, siis laseb sobitamise töötleja sul uue võre vana võre külge kleepida, justkui prooviksid sa seda võimalikult hästi peale sobitada.

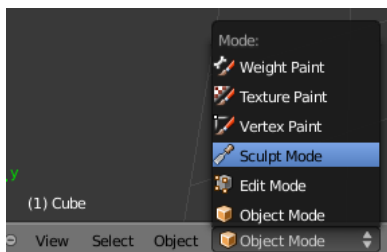
Ülevaade

Voolimisrežiim Sculpt on sarnane muutmisrežiimile Edit, kuna selle kaudu saab muuta mudeli kuju, kuid voolimisrežiimis töötamine on kõvasti teistsugune: selle asemel et tegeleda üksikute elementidega (tipud, servad ja küljed), muudetakse mingit mudeli ala pintsliga. Teisisõnu, selle asemel et valida grupp tippe, valib voolimisrežiim tipud vastavalt sellele, kus asetseb pintsel, ning muudab neid.

Voolimisrežiim

Voolimisrežiim valitakse režiimi menüüst 3D-vaate (3D View) päises.

Kui voolimisrežiim on sisse lülitatud, muutub 3D-vaate tööriistariba režiimile vastavateks paneelideks. Tööriistariba paneelid on pintsel (Brush), tekstuur (Texture), tööriist (Tool), sümmeetria (Symmetry), tõmme (Stroke), kõver (Curve), välimus (Appearance) ja valikud (Options). Samuti ilmub punane ring, mis järgib 3D-vaates kursori asukohta.



Voolimisrežiimi rippmenüü.

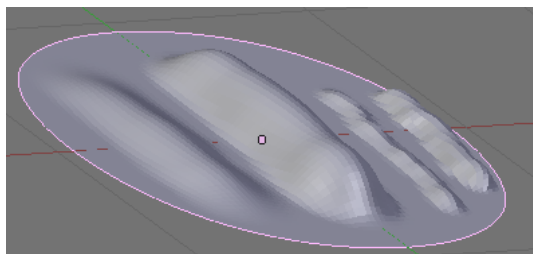


Kursor
voolimisrežiimis.

Voolimise pintslid

Pintslid ja pintslite eelvalikud. Need on kombineeritud 'tööriistad', millel on kindel tõmme, tekstuur ja muud valikud.

Voolimisrežiimis on kuusteist pintslit ning igaüks neist muudab mudelit omalaadsel viisil. Mitmeid neist saab määrata kasutama lisavat või eemaldavat efekti. Neid saab valida tööriistamenüüst Tool. Praegused pintslite eelvalikud on: mull, savi, viik, joonista, täida/süvenda, lamenda/toonita, haara, paisuta/tõmba kokku, kiht, nihuta, pigista/suurenda, pööra, kraabi/lükka, silu, madu, põial:



Erinevate suuruste ja tugevustega joonistamine.

Blob (mull)

Surub võret keraja vormiga välja- või sissepoole.

Clay (savi)

Samane joonistamise pintsliga Draw, kuid sellel on sätted, millega saab määrata tasapinna, millel pintsel töötab.

Crease (viik)

Loo võret tõugates või tõmmates teravaid süvendeid või valle, samal ajal tippe üksteisega kokku pressides.

Draw (joonista, D)

Liigutab tippe sisse- või väljapoole sõltuvalt joonistatud pintslitõmbe sisse jäävate tippude keskmisest normaalist.

Fill (täida)

Töötas samamoodi nagu lamendamise pintsel, kuid toob ülespoole ainult need tipud, mis jäävad pintslit tasapinnast allapoole. Täitmise vastand on süvendamine Deepen, mis lükkab tasapinna alla jäävaid tippe allapoole.

Flatten (lamenda, T)

Lamendamise pintsel leiab 'pinnatasandi', mis asub vaikumisi pintslit ala sisse jäävate tippude keskmise kõrguse kohal. Tipud tõmmatakse seejärel sellele pinnale lähemale. Lamendamise vastand on toonitamise pintsel Contrast, mis lükkab tippe pintslit tasandist üles- või allapoole eemale.

Grab (haara, G)

Haaramist kasutatakse punktidegrupi lohistamiseks. Erinevalt teistest pintslitest ei muuda haaramine erinevaid punkte, kui pintslit mööda mudelit tõmmata. Selle asemel valib haaramine hiireklahvi vajutuse peale grupi tippe ning tõmbab neid hiirega kaasa. See efekt töötab samamoodi nagu muutmisrežiimis Edit sisselülitatud proportsionaalse muutmisega tippudegrupi lohistamine, kuid haaramine saab kasutada ka teisi voolimisrežiimi võimalusi (nagu tekstuurid ja sümmeetria).

Inflate (paisuta, I)

Samane joonistamise pintsliga Draw, kuid paisutamise pintsel liigutab tippe ringi vastavalt nende endi normaalidele.

Layer (kiht, L)

Samane joonistamise pintsliga Draw, kuid ümberpaigutamise kihi kõrgus on piiratud. Tekitab mulje, nagu joonistataks objekti pinnale ühtlase paksusega tahket kihti. See pintsel ei joonista samal tõmbel iseenda peale; tõmme ristub iseendaga samal kõrgusel. Hiireklahvi lahtilaskmine ja uue tõmbe alustamine algseadistab sügavuse ning siis maalitakse eelmise tõmbe peale.

Nudge (nihuta)

Liigutab tippe pintsli tõmbe suunas.

Pinch (pigista, P)

Tõmbab tippe pintsli keskkoha suunas. Vastupidine säte on suurendamine (Magnify), mille puhul lükatakse tipud pintsli keskkohast eemale.

Rotate (pööra)

Pöörab pintsli sees tippe vastavalt kursori liikumise suunale.

Scrape (kraabi)

Pintsel töötab nagu lamendamise oma, kuid lükkab allapoole ainult tasapinnast kõrgemale jäävaid tippe. Kraapimise vastand on lükkamine (Peak), mis lükkab tasapinnast kõrgemal olevaid tippe pinnast kõrgemale.

Smooth (silu, S)

Nagu nimigi ütleb, vähendab see pintsli mõjualas oleva võre ebatasasusi, siludes tippude asukohti.

Snake Hook (madu)

Tõmbab tippe pintsli liikumisega kaasa, luues pikki maolaadseid vorme.

Thumb (pöial)

Samane nihutamise pintsliga Nudge, kuid see lamendab pintsli alas olevat võret, liigutades seda samas pintsli tõmmete suunas.

Mitmikresolutsiooni töötlejaga voolimine (*Multires Modifier*)

...

Voolimise omaduste paneel

See paneel ilmub tööriistakasti 3D-vaateakna vasakul küljel.

Pintsli menüü (*Brush*)

Radius (raadius)

See säte määrab pintsli raadiuse pikslites. 3D-vaates saad kiirvaliku F kaudu muuta pintsli suurust interaktiivselt, lohistades hiirt ja vajutades seejärel vasakut hiireklahvi (pintsli tekstuur peaks ringi sees nähtav olema). Sedamoodi suurust muutes saad numbri sisestamise ja Enteri vajutamisega määrata suuruse numbriliselt. Pintsli suurust saab mõjutada ka survetundlikuse ikooni sisselülitamise läbi, kui kasutad sobivat puutelauda.

Strength (tugevus)

Tugevus määrab, palju iga pintsli tõmme mudelit mõjutab. Näiteks kõrgemad väärtused panevad joonistamise pintsli Draw kiiremini mudelile sügavust lisama ning silumispintsli Smooth mudelit kiiremini siluma. Seda sätet ei saa kasutada haaramise (Grab), mao (Snake Hook) või pööramise (Rotate) pintsli puhul.

Kui tugevuse säte ei näi mudeli puhul sobivat (näiteks ka kõige madalam tugevus tekitab ikkagi liiga suuri muutusi), siis võid mudeli mõõtkava suurendada (muutmisrežimis Edit, mitte objektirežimis Object). Suurem objekti suurus muudab pintsli mõju väiksemaks ja vastupidi. Pintsli tugevust saab ka interaktiivselt muuta, vajutades 3D-vaates \diamond ShiftF, liigutades seejärel pintsli ja vajutades vasakut hiireklahvi. Ka seda suurust muutes saad suuruse sisestada numbriliselt. Pintsli tugevust saab mõjutada ka survetundlikuse ikooni sisselülitamise teel, kui kasutad sobivat puutelauda.

Autosmooth (automaatne silumine)

Sisselülitamisel rakendatakse iga tõmbe puhul automaatset silumist.

Sculpt Plane (voolimise tasapind)

Kasuta seda menüüd, et määrata tasapind, millel voolimine toimub.

Plane Offset (tasapinna nihe)

Sätib tasapinda, millel pintsel kas vaataja suunas või vaatajast eemale töötab.

Trim (tasapinna piiramine)

Lülitab sisse voolimistasapinna piiramise, määratuna kauguse välja Distance väärtusest, mille puhul pintsli mõju kaob.

Front Faces Only (ainult esiküljed)

Sisselülitatuna mõjutab pintsel ainult vaataja poole suunatud tippe.

Accumulate (üksteise peale)

Sunnib pintsli tõmbe mõju akumuleeruma enda peale.

Tõmbe menüü (*Stroke*)

Stroke Method (tõmbe meetod)

Määrab, kuidas pintsli tõmbeid võrele lisatakse:

Dots (punktid)

Tavalised pintsli tõmbed.

Drag Dot (lohisev punkt)

Pintsli kuju tekitab ühe nihke. Vajuta hiireklahvi ja lohista võre soovitud kohta ning lase seejärel klahv lahti.

Space (punktiir)

Pintsli tõmme tekitab punktide seeria, mille vahelise kauguse määrab vahemaa väärtus Spacing. Spacing tähistab protsenti pintsli diameetrist.

Anchored (ankurdatud)

Tekitab nihke pintsli asukohas. Hiireklahvi vajutamine ja lohistamine muudab pintsli diameetrit. Kui sisse on lülitatud säte 'servast servani' Edge to Edge, määrab pintsli asukoha ja suuna kahe punktiga määratud ring, mille puhul esimene hiirevajutus määrab esimese ning lohistamine teise punkti, mis asub ringil esimese vastas.

Airbrush (pihusti)

Pintsli mõju kestab seni, kuni hiirt all hoitakse, ning selle mõju määrab säte Rate. Kui see on välja lülitatud, mõjutab pintsel mudelit ainult asukoha muutmisel. Haaramispintsliga Grab seda meetodit kasutada ei saa.

Järgnevaid parameetreid saab kasutada punkt- (Dots), punktiir- (Space) ja pihustimeetodite (Airbrush) puhul:

Smooth stroke (sujuv tõmme)

Pintsel jääb hiirest maha ja järgib sujuvamat rada. Sisselülitamisel saab muuta järgnevaid seadeid:

Radius (raadius)

Määrab minimaalse vahemaa eelmisest punktist, enne kui tõmme jätkub.

Factor (faktor)

Määrab silumise hulga

Jitter (võdin)

Võdistab maalimise ajal pintsli asukohta.

Kõvera menüü (*Curve*)

Kõvera sektsiooni Curve kaudu saad kasutada kõverat, mis määrab pintsli intensiivsuse erinevuse keskosa (kõvera vasak pool) ja äärte vahel (kõvera parem pool).

Tekstuuri menüü (*Texture*)

Pintsli tugevuse määramiseks saab kasutada ka tekstuur. Vali tekstuurikastist mõni olemasolev tekstuur või loo nupuga New uus.

Brush Mapping (pintsli laotus)

Määrab, kuidas tekstuur pintsli tõmbele laotatakse:

Fixed (fikseeritud)

Kui see on valitud, järgib tekstuur hiirt ning tekib mulje, nagu lohistataks tekstuuri üle mudeli.

Tiled (korduv)

Kui see on valitud, korratakse tekstuuri üle ekraani ning tekib mulje, nagu liiguks pintsel tekstuurist eraldi. See valik on kõige kasulikum korduvate piltide ja mitteprotseduuriliste tekstuuride puhul.

3D

See valik lubab pintsli kasutada kõiki protseduuriliste tekstuuride võimalusi. Kasutatava tekstuuri koha määramiseks kasutatakse pintsli asukoha asemel tippude koordinaate.

Angle (nurk)

See on tekstuuripintsli pöördenuhk. Seda saab 3D-vaates valikuga CtrlF interaktiivselt muuta. Interaktiivse pööramise ajal saad väärtuse sisestada ka numbriliselt. Võimalikud valikud on:

User (kasutaja)

Määra nurga väärtus otse.

Rake (riisu)

Nurk sõltub pintsli tõmbe suunast. Seda ei saa 3D tekstuuride puhul kasutada.

Random (juhuslik)

Nurk on juhuslik.

Offset (nihe)

Täpsustab tekstuurilaotuse asetust X-, Y- ja Z-teljel.

Size (suurus)

Selle kaudu saad määrata tekstuuri mõõtkava. Lohistatavate tekstuuridega (Drag) seda kasutada ei saa.

Sample Bias (sämpli nihe)

Tekstuurisämplitele lisatav väärtus.

Overlay (pealiskiht)

Sisselülitamisel näidatakse tekstuuri vaateaknas tekstuuri vastavalt alfa väärtusele Alpha.

Sümmeetria menüü (*Symmetry*)

Peegelda pintsli tõmbeid üle valitud kohaliku telje. Pea meeles, et kui soovid muuta telgede suunda, pead mudelit pöörama muutmisrežiimis Edit, mitte objektirežiimis Object.

Feather (pehmenda)

Vähendab pintsli tugevust, kui see ületab sümmeetria telje joont.

Radial (ringikujuline)

Nende sätetega saad tekitada ringikujulist sümmeetriat ümber soovitud telje. Number määrab, mitu korda tõmmet 360 kraadi jooksul ümber soovitud keskse telje korratakse.

Valikute menüü (*Options*)

Hargtöötusega voolimine (Threaded Sculpt)

Kasutab voolimise jõudluse parandamiseks kõiki protsessori tuumasid.

Fast Navigation (kiire navigeerimine)

Mitmikresolutsiooniga (Multires) mudelite puhul näidatakse vaateaknas liikumise ajal madalama resolutsiooniga varianti.

Show Brush (näita pintslit)

Näitab vaateaknas pintslit.

Seotud sätted

Size (suurus)

Sunnib erinevaid pintsleid kasutama sama suurust.

Strength (tugevus)

Sunnib erinevaid pintsleid kasutama sama tugevust.

Lock (lukustus)

Nende kolme nupu abil saad sa voolimise ajal piirata oma mudeli muutmist/moondamist piki kohalikku telge.

Välimuse menüü (*Appearance*)

Saad määrata pintslite värvuse vastavalt sellele, kas see on liitmise või lahutamise režiimis.

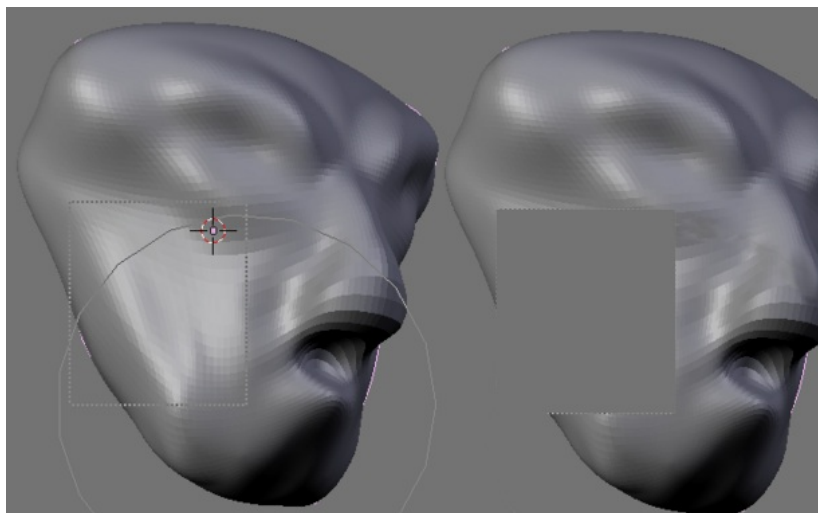
Samuti saad määrata pildifailiga pintslite ikooni.

Tööriista menüü (*Tool*)

Siit saad valida pintslite eelseadistuse. Algseadistamise nupp Reset Brush taastab pintslite vaikesätted. Samuti saad sundida Blenderit lülitite abil valitud pintslit kasutama tippude maalimise (Vertex Paint), kaalude maalimise (Weight Paint) ja tekstuuri maalimise (Texture Paint) režiimides.

Võre peitmine ja näitamine

Mõnikord on väga kasulik eraldada voolimiseks mõni võre osa. Võre osaliseks peitmiseks lohista kombinatsiooniga Ctrl+Shift LMB ümber ala, mida soovid näha - ülejäänud peidetakse. Kasuta kombinatsiooni Ctrl+Shift RMB, kui soovid peita ainult valitud nelinurka. Kõigi peidetud alade nähtavaks muutmiseks vajuta AltH või kasuta korra kombinatsiooni Ctrl+Shift LMB.



Enne ja pärast peitmist.

Klaviatuuri kiirvalikud

Tegevus

Peida väljaspool valikut olev võre

Peida valikus olev võre

Näita tervet võret

Lülita pihusti sisse/välja

Muuda interaktiivselt pintslite suurust

Kiirvalik

Ctrl+Shift LMB

Ctrl+Shift RMB

AltH

A

F

Muuda interaktiivselt pintsli tugevust	⇧ ShiftF
Pööra interaktiivselt pintsli tekstuuri	CtrlF
Vaheta pintsli suunda (liida Add/lahuta Sub)	V
Joonistamise pintsel Draw	D
Silumise pintsel Smooth	S
Pigistamise pintsel Pinch	P
Paisutamise pintsel Inflate	I
Haaramise pintsel Grab	G
Kihi pintsel Layer	L
Lamendamise pintsel Flatten	T
X-sümmeetria	X
Y-sümmeetria	Y
Z-sümmeetria	Z
Lülita voolimise paneel sisse/välja	N
Mine ühe mitmikresolutsiooni taseme võrra üles	Page up
Mine ühe mitmikresolutsiooni taseme võrra alla	Page down

Tipugrupid (Vertex Groups)

Võre koosneb servade abil tervikuks ühendatud punktidest ehk tippudest. Keerulisemate objektide korral võib tippe olla tuhandeid (või isegi miljoneid). Blender võimaldab nende grupeerimist järgmistel põhjustel:

- Punktide taaskasutus võre osade kopeerimiseks
- "Üleliigsete" punktide peitmiseks, et saaks segamatult soovitud punkte töödelda.
- Dokumenteerimiseks ja teistele selgituste andmiseks.
- Objekti deformeerimiseks skeleti abil.
- Osakeste genereerimiseks mingist kindlast punktide kogumist.
- Genereeritud osakeste kiiruse defineerimiseks.
- Ühe objekti küljes mitme materjali kasutamiseks

Skeletid

Punktide grupe saab iga luu jaoks automaatselt tekitada. Seda küllaltki keerukat toimingut kirjeldatakse skeleti elustamisega seotud peatükis. Selles teemas on põhirõhk modelleerija enda poolt loodud tipugruppidel.

Mille poolest on tipugruppide kasutamine hea?

Tipugruppide abil saab defineerida objekti alamkomponente nagu näiteks tooli jalad või uksehinged. Objekti osade kallal on kergem tööd teha siis, kui nende osad on tipugruppideks jaotatud. Nii saab teha tööd ühe kindla grupiga ilma seda objektist eraldamata. Peitmise käsuga saab kõik teised grupivälised punktid nähtamatuks muuta, et need ei segaks parajasti käsiloleva grupi töötlemist.

Tipugrupid hõlbustavad ka võre osade kopeerimist ja korduvkasutamist. Võtame näiteks Lego klotsi modelleerimise. Lihtsaim klots koosneb põhikehast ja kinnitusnagast. Näiteks nelja kinnitusnagaga klotsi modelleerimisel tuleb valida naga tipugrupi tipud ja endiselt muutmisrežiimis olles nendest koopia teha ning seejärel uude kohta nihutada.

Veel leiavad tipugrupid kasutust skeleti sidumisel võrega. Tegelase animeerimiseks tuleb teha terve hunnik tipugruppe ning seejärel skelett. Renderdaja jaoks nähtamatute luude abil saab siis tipugruppe liigutada. Luu liigutamisel liiguvad tipud kaasa. Mitte kõik, vaid ainult need, mis on vastava luuga seotud. Kui nii tipugrupi kui ka luu nimi on "Käsi", siis luu "Käsi" liigutamisel liiguvad tipud, mis kuuluvad gruppi "Käsi". Tipugruppi "Jalg" kuuluvad tipud aga luu "Käsi" liigutamisel kaasa ei liigu. Nii deformeeruvad ainult teatud osad võrest, samal ajal kui teised osad jäävad paigale.

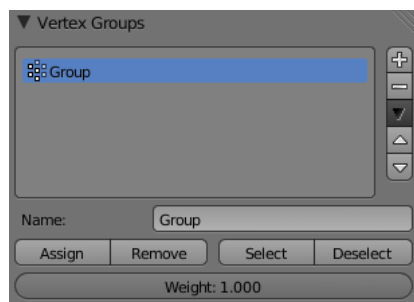
Osakeste menüüs Particle System on tipugruppide paneel Vertex group, mille kaudu saab mitmeid asju piirata läbi tipugruppide kasutamise. Tipugruppide kaalude maalimise teel saab määrata osakeste hulka, suurust ja kiirust. Näiteks osakeste genereerimisel tüübiga "juuksed" (Hair) saab määrata, et juuksed tekiksivad vaid nendesse kohtadesse, kus tipud kuuluvad vastavasse gruppi.

Ka mitmed töötlejad kasutavad mõju piiramiseks tipugruppe. Mõningate töötlejate toimima hakkamiseks on tingimata tarvis tipugruppe. Näiteks [masktöötleja \(Mask Modifier\)](#) puhul.

Märkus

On mitmeid omadusi, mille määramiseks mingile piirkonnale pole tipugruppide tekitamine tarvilik. Mitme materjali kasutamisel ühe objekti juures tuleb vastavad tipud materjaliga seostada, aga selleks pole vaja teha tipugruppe.

Tipugruppide loomine ja kustutamine

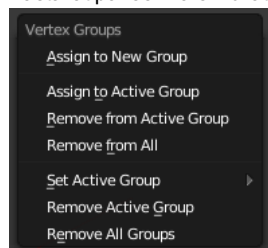


Tipugruppide paneel Vertex Groups muutmisrežiimis

Vaikimisi pole ühelgi objektil mitte ühtegi tipugruppi ja kõik tipud on n-ö. üksikud ja omaette. Kuigi tipugruppe saab teatud juhtudel automaatselt genereerida, näiteks kaalude maalimisel, loome me enamasti tipugrupid käsitsi tipugruppide paneeli Vertex group abil objekti andmete menüü Object data all. Kui tipugrupp on lisatud ja me oleme muutmisrežiimis (Edit Mode), kuvatakse tipugruppide paneelil vajalikud nupud.

Tipugrupid on olemas ainult võreobjektidel.

Tipugrupid on olemas ainult nendel objektidel, millel on olemas tipud. Näiteks tekstiobjektid ei koosne tippudest ja nende puhul vastavat paneeli ka ei kuvata. Tipugruppe kuvatakse ainult tippudest koosnevate objektide puhul.



Tipugruppide hüpikmenüü

Vertex Groups

Tipugrupi tekkitamiseks vajuta vasema hiireklahviga + nupul. Selle tegevuse tulemusena tekib uus tipugrupp (mille nimi on - üllatus-üllatus - "Group") ja ilmub nähtavale ka kaalude määramise Weight liugur. Senise tegevuse tulemusena veel ükski tipp sinna gruppi ei kuulu. Et märgitud tipud ka sinna gruppi kuuluma hakkaksid, tuleb vajutada nuppu Assign. Kui tahad valitud tipud ühe liigutusega uude tipugruppi saada, siis kasuta klahvikombinatsiooni CtrlG.

Kontrolli oma loodud seosed üle!

Veendumaks, et tipud on ikka õiges grupis, saab kasutada valimise ja valiku tühistamise nuppe Deselect ja Select. Kui selle peale midagi ei juhtu, siis tee soovitud tipud märgituks ja vajuta nuppu Assign.

Tipugrupi eemaldamiseks kasuta - nuppu. Nii labane see ongi. Iga sinna gruppi kuulunud tipp on nüüd tollest grupist välja arvatud. Tuleb siiski mees pidada, et tipud võivad kuuluda mitmesse gruppi. Ühest grupist eemaldamine ei mõjuta selle kuulumist mõnda teise gruppi.

Tipu nimeks tuleks panna midagi loominguilisemat, kui seda on "Group". Selleks klõpsa nimeväljal vasempoolse hiireklahviga ja trüki midagi uut. Asja selgitavate nimede kasutamine on suurte tööde ja paljude inimeste koostöö puhul äärmiselt tähtis.

Valimine ja valiku tühistamine

Praktika näitab, et enne tippude lisamist gruppi või sealt eemaldamist on kasulik kontrollida, millised tipud juba millisesse gruppi kuuluvad. Selleks tühistada esimese asjana kõikide tippude valik, kasutades üks või kaks korda järjest A klahvi, olles samal ajal hiire kursoriga 3D-vaate alal. Siis klõpsa hiirega soovitud grupi nimel ja vajuta nuppu Select. 3D-vaate aknas muutuvad seejärel valituks sinna gruppi kuuluvad tipud.

Mõnikord on vaja kindlaks teha, millised tipud ei kuulu ühtegi gruppi. Selleks vali esmalt 3D-vaates olles kõik tipud klahviga A või kui esmalt juba oli midagi valitud, siis vajuta seda klahvi kaks korda järjest. Nüüd klõpsa vasema hiireklahviga esimese grupi nimel ja vajuta nuppu Deselect, et sinna kuuluvad tipud valikust eemaldada. Jätka protseduuri kõigi gruppidega. Kui oled nii kõik grupid läbi käinud, siis järelejäänud tipud on üksikud. Umbes nagu pallimängu jaoks meeskondade valimise puhul.

Tippude tipugruppi lisamine ja eemaldamine

Tipu lisamiseks tipugruppi tee ühte järgnevatest:

1. Vali tipugruppide nimekirjast soovitud tipugrupp.
2. Kasuta tippude märkimiseks hiire parempoolset klahvi.
3. Märgitud tippude tipugruppi lisamiseks vajuta nuppu Assign või kasuta klahvikombinatsiooni CtrlG, mille järel avaneb mitmesuguste seonduvate valikutega menüü.

Pea mees, et tipud võivad samaaegselt kuuluda mitmesse erinevasse gruppi.

Märkus

Assign nupp lisab valitud tipud aktiivsesse gruppi. Gruppi juba kuuluvaid tippe selle tulemusel grupist ei eemaldata.

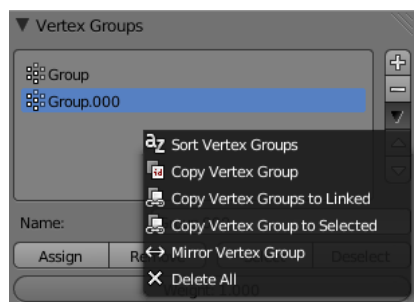
Tippude eemaldamine grupist:

1. Vali gruppide nimekirjast soovitud grupp.
2. Vali tipud, mida sa soovid grupist eemaldada.
3. Klõpsa vasema hiireklahviga Remove nupul või kasuta klahvikombinatsiooni CtrlG, mille tulemusel avaneb seda võimaldava valikuga hüpikmenüü.

Märkus

On võimalik ka kõik valitud tipud kõikidest gruppidest korraga eemaldada, kui kasutada klahvikombinatsiooni CtrlG ja valida avanenud hüpikmenüüst (CtrlG » Remove from All).

Tipugruppide haldamine



Haldamist võimaldav rippmenüü

Sort Vertex Groups (sorteeri tipugruppe)

Sorteerib grupid tähestikuliselt

Copy Vertex Group (kopeeri tipugruppe)

Teeb olemasolevast tipugrupist koopia (uus koopia sisaldab ka kõiki originaalgrupis olnud tippe)

Copy Vertex Group to Linked (kopeeri tipugrupid linkideks)

Kopeerib tipugrupi kõikidele lingitud duplikaatidele

Copy Vertex Group to Selected (kopeeri tipugrupid valitutesse)

Kopeerib tipugrupi teiste valitud võrede jaoks

Mirror Vertex Group (peegelda tipugruppi)

Peegeldab kõik tipugrupid, vahetab kaalud ja/või nimed, muudab ainult märgitud tippe, mõlema poole märkimisel vahetab, muul juhul kopeerib kaalud märkimata poolelt.

Delete All (kustuta kõik)

Kustutab kõik tipugrupid

Tipu kaalude maalimise režiim (*Weight Paint Mode*)

Tipu kaalude maalimist (Weight Painting) kasutatakse tipugruppide (Vertex Group) loomiseks ja muutmiseks. Tipul on lisaks tipugruppi kuulumisele ka gruppi kuulumise kaal. Tipu kaal (*weight*) tähistab tipugrupi mõju tipule edasikandumise tugevust. Väärtus "1" tähistab 100% mõju ja "0" olematut mõju, st tipugrupi mõjutamine (näiteks mõne töötlejaga) tippu ennast ei mõjuta.

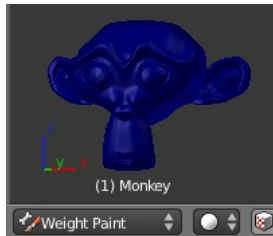
Põhiliselt kasutatakse kaalude maalimist võrede taageldamisel (*rigging*). Lisaks kasutatakse seda veel osakeste generaatori ja karvade tekitamise juures.

Mode: Kaalude maalimise režiim Weight Paint

Hotkey: Ctrl⇧ Tab

Menu: Mode menüü

Kui sa lülitud kaalude maalimise (Weight Paint) režiimi (ja pole veel ühtki gruppi valinud), näed sa objekti kergelt varjutatud sinisena (**Pilt 2**).

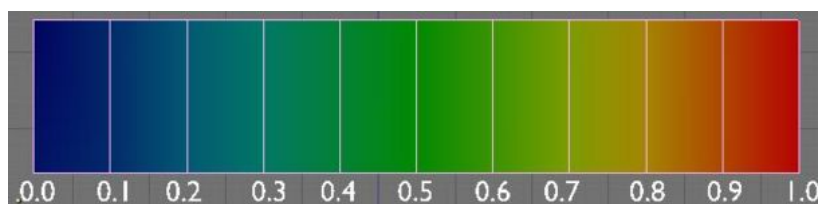


Pilt 2: objekt kuvatuna kaalude maalimise režiimis Weight Paint.

Värvi abil visualiseeritakse iga tipu kaal aktiivses grupis. Sinisena kuvatud tipp tähendab ühte järgnevatest: selle kaal on null, see ei kuulu valitud gruppi või et see ei kuulu mitte ühessegi gruppi.

Kaale indikeerivaid värve saad ise valida kasutaja eelistuste kaudu: User Preferences->System->Custom Weight Paint Range.

Tipule mõju andmiseks tuleb võret vastava värviga maalida. Kui ühtki tipugruppi ei eksisteeri või ükski pole aktiivne, siis mõjuala maalimisega alustamine tekitab automaatselt ühe uue tipugrupi. Kui maalitav tipp ei kuulu ühessegi aktiivsesse gruppi, siis maalimise tulemusena lisatakse see aktiivsesse gruppi. Seda isegi sellisel juhul, kui mõjualaks on määratud null. Kasutatud värvispekter on kuvatud pildil nr 3.



Pilt 3: kasutatavad värvid ja neile vastavad kaalud.

Võrele maalimist saab teha pintsliga. Värv annab edasi ainult tipu kaalu. See ei ole seotud servade ega külgedega. Nii et nende maalimist pole mõtet üritada. Pintsli tööriista valikud ilmuvad tööriistariulile. Selle avamiseks kasuta klahvi T või vali menüüst *View > Tool Shelf*.

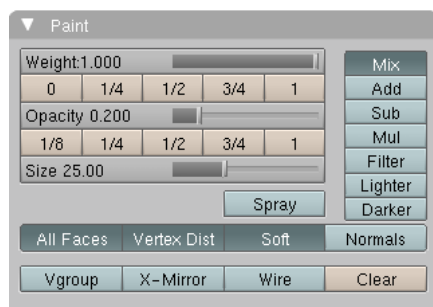


Ellujäämise nipid mõjuala maalimisel

Mõningad kaalude maalimist lihtsustavad näpunäited:

- Pintsli suurendamiseks vajuta klaviatuuril F klahvi, liiguta hiirt soovitud suunas ja klõpsa vasemat hiireklahvi.
- Ainult kindla suurusega kastikujulisele alale maalimiseks vajuta AltB ja lohista hiirega üle soovitud ala. Tulemuseks on see, et ülejäänud ala enam ei näidata ja maalimine mitternähtavat ala ei mõjuta. Maalida saab ainult nähtavale alale. Alast väljumiseks vajuta klahvikombinatsiooni AltB.

Maalimise paneel



Pilt 4: maalimise paneel Paint muutmise kontekstis Editing.

Maalimise riista valikud on keerulised ja kaalude maalimine on võimalik kõige peenemate nüanssideni. Tavaliselt pole kõiki neid keerulisi variante tarvis ja kaalude maalimiseks on vaja neist vaid mõningaid. Enimkasutatavad seadistused on näidatud **paksus kirjas**.

Weight (kaal)

Millist kaalu (värvi) maalimisel kasutada. Pintselduse kujulisele nupule vajutades avaneb rida eelseadistatud pintsleid. Pintsel võib olla seadistatud töötama absoluutse väärtusega. Näiteks kui "kaaluks" on määratud 0,2, siis maalimisel saabki tipp väärtuseks 0,2 - endine väärtus tulemust ei mõjuta.

Auto Normalize (automaatne normaliseerimine)

Tingib selle, et maalimise tulemusel oleks tipu kaalude kogusumma kõigi tipugruppide peale alati 1.

Radius (raadius, F)

Maalispintsli raadius, mida kujutatakse ringina.

Strength (tugevus)

Kui suurt mõju pintsel maalitavale alale avaldab.

Jitter (võbelus)

Pintsli asukoha varieerimine maalimisel.

Kaalude tööriistad (*Weight Tools*)

Nende tööriistade aktiveerimisel ilmuvad veel mõningad valikud maalimise omaduste paneelile.

Normalize All (normaliseeri kõik)

Muudab tipu kaale kõigis tipugruppides, nii et nende kogusumma oleks võrdne ühega. See tööriist mõjutab kõiki tipugruppe.

Lock Active (lukusta aktiivne)

See hoiab aktiivse tipugrupi kaalud muutumatuna ja normaliseerib teised tipugrupid.

Normalize (normaliseeri)

See tingib ainult aktiivse tipugrupi normaliseerimise

Invert (tagurpidi)

Keerab tippude kaalude väärtused pahupidi

Add Weights (lisa kaalud)

Lisab enne pahupidi pööramist nullkaaluga tipud

Remove Weights (eemalda kaalud)

Eemaldab enne pahupidi pööramist nullkaaluga tipud

Clean (puhasta)

Eemaldab mittevajalikud tipugrupi seosed (tipu kaal tipugrupis on null)

Limit (piira)

Eemaldab määratud väiksema kaaluga tipud tipugrupist

All Groups (kõik grupid)

Korrastab kõik tipugrupid

Keep Single (hoia lihtsana)

Jätab koristamise ajal iga tipu vähemalt ühte gruppi

Levels (tasemed)

Timmib kaalude väärtuseid järgmiste tööriistadega:

Offset (nihe)

Lisab väärtuse olemasolevale kaalule

Gain (tõus)

Korrutab kaalu määratud väärtusega

Pintslitõmbed (*Stroke*)

Airbrush (aerograaf)

Airbrush seade jätkab maalimise efekti hiire all hoidmisel.

Smooth Stroke (sujuv tõmme)

See tingib pintsli kursorist mahajäämise ja trajektoori ühtlustamise. Nii saab sujuvama joone.

Radius (raadius)

Väikseim kaugus viimasest punktist, enne kui joon jätkub

Factor (faktor)

Suurem väärtus annab sujuvama joone

Space (vahemaa)

Selle abil saab "täpiliselt" maalida. Spacing suurus määrab ära, kui pika vahemaa tagant üks "täpp" tuleb.

Mõjukõvera kuju (*Curve*)

Selle abil saad luua omaloodud mõjukõveraga pintsli

Alumise rea nupudelt saad kätte eelseadistatud kõverad.

Valikud

All faces (kõik küljed)

Selle sisselülitamisel toimib maalimine ainult tippudele, mis kuuluvad külgedele, millele kursor parasjagu osutab. See valik on kasulik siis, kui tegemist on keerulise võrega. Siis saab vältida vaates näiliselt lähedaste, aga tegeliku sügavuse suhtes kaugete osade maalimist.

Normals (normaalid)

Tipu normaal aitab määrata maalimise ulatust. See on umbes selline efekt nagu maaliks valgusega.

Spray (pihusti)

Spray akumulēerib kaalu väärtused iga hiireliigutuse peale.

X-mirror (X-pegeldus)

Selle abil saab mõjuala sümmeetriliselt maalida ja seda juhul, kui tipugrupid on sama nime ja poolt näitava laiendiga. Näiteks `.R/.L` või `_R/_L` (parem/vasak). Kui peegeldav grupp puudub, toimub sümmeetriline maalimine grupi enda sees. Nimetamise tingimuste kohta vaata täpsemalt teemast [Skeleti muutmine: nimetamise reeglid](#). Samad tingimused kehtivad ka skeletile/kontidele.

Topology Mirror (topoloogia peegeldamine)

Topoloogial põhinevat peegeldamist võib kasutada siis, kui võre on sümmeetriline.

Ühtlustatud seaded: Pintsli suurus (Size) ja tugevus (Strength) võib olla mitme eelseadistatud pintsli jaoks ühine või ka igaühe jaoks erinev.

Väljanägemine (*Appearance*)

Siit saad määrata pintsli kursori värvi või laadida oma kursori ikooni välisest failist.

Tööriist (*Tool*)

Sellega saab määrata, kuidas tippude kaale mõjutatakse.

Mix (segamine)

Uus kaal asendab (järk-järgult) vana kaalu. Uue kaaluga korduval maalimisel saab tipu kaaluks lõpuks pintsli kaal.

Add (lisamine)

Toimub vanale kaalule uue juurdeliitmine. Kaalude maalimisel tuleb alati mõelda kaalude liitmisele, mitte aga värvidele: millelegi sinise (**0.0**) lisamine ei muuda midagi, aga rohelise (**0.5**) lisamine rohelisele annab tulemuseks punase (**1.0**), ehk puhas matemaatika, mitte värviteooria.

Sub (lahutamine)

Toimub pintsli kaalu mahalahutamine olemasolevast. Jällegi tuleb mõelda kaalude väärtustele, mitte värvidele.

Mul (korrutamine)

Vana kaal korrutatakse uuega.

Blur (hajutamine)

Keskmistab maalitavaid kaale omavahel.

Lighten (helendamine)

Maalib ainult selliseid tippe, mis on valitud kaalust väiksema kaaluga.

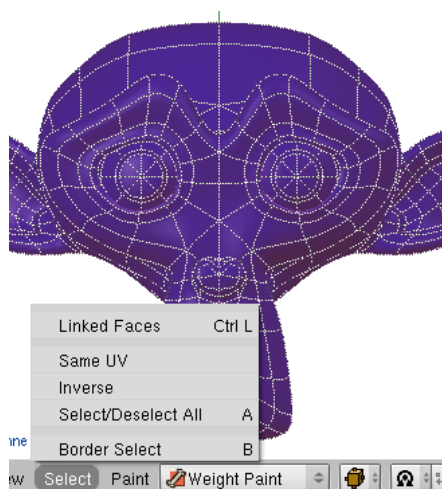
Darken (tumendamine)

Maalib ainult selliseid tippe, mis on valitud kaalust suurema kaaluga.

- **Vgroup (tipugrupp):** Toimub ainult aktiivsesse tipugruppi kuuluvate tippude maalimine. See on väga kasulik selleks, et teisi tipugruppe segi ei aetaks.

Külgede valiku maskimine (*Face Selection Masking*)

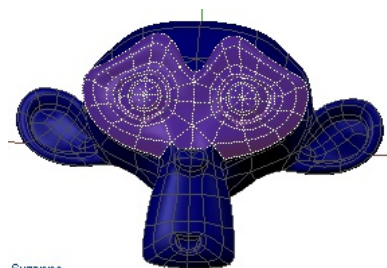
Keerulise võre korral on mõjukuse maalimise režiimis mõningatele osadele suht võimatu ligi pääseda. Väga raske on aru saada, kus kohas täpselt mingi tipp asub. Selle probleemi lahendamiseks on hea ja lihtne meetod: režiim Face Selection Masking (külgede valiku maskimine). 3D-vaate alumises servas oleval nupureal on nupp Face Selection masking, mille abil saab märkida ainult need küljed, mille hulka kuuluvate tippude maalimist parajasti tahetakse teha..



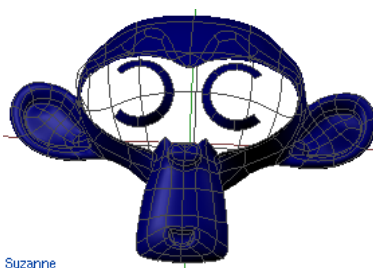
Pilt 5: valimise menüü Select kaalude maalimise režiimis Weight Paint.

Valimise (Select) režiimil on tavalise kaalude maalimise (Weight Paint) režiimi ees eeliseid:

1. Ka pinnatükeldaja kasutamisel kuvatakse vaid originaalvõret. On näha ülemaalimiseks võimaldatud tipud.
2. Võid valida ka küljed, maalitakse ainult valitud külgede tippe.
3. Valimise tööriistad on sellised:
 - RMB – üksiku külje valimine. ◊ Shift RMB mitme külje valimiseks.
 - A – valib kõik küljed või teeb valiku olemasolul kõik mittevalituks.
 - B – kastikujulise piirkonnaga valimine.
 - BB – ringikujulise alaga ehk pintsliga valimine.
 - CtrlL – lingitud osade valimine.
 - Valiku pahupidi pööramiseks kasuta klahvikombinatsiooni CtrlI.
4. Valitud külgede peitmiseks on klahv H ja uuesti nähtavale toomiseks AltH (**Pilt 6**).



Pilt 6a: Kattuvate külgede valimine...



Pilt 6b: ... ja nende peitmine klahviga H.

- Ühe piirkonna eraldamiseks muust segavast ümbritsevast võib kasutada lõikavat raami (*Clipping Border*). Selle tekitamiseks vajuta klahvikombinatsiooni AltB ja lohista vasemat hiireklahvi all hoides üle soovitud ala. Ülejäänud osa 3D-vaatest on nüüd peidetud. Peidetud piirkonna peidust väljatoomiseks kasuta uuesti klahvikombinatsiooni AltB.

Kaalude maalimine skeleti elustamisel

Sellel otstarbel kasutatakse kaalude maalimist ilmselt kõige sagedamini. Luu liikumisel peavad sellega kaasa liikuma ka samas piirkonnas olevad võre tipud, kuid kõik ei pea liikuma täielikult - et tekiks mulje liigese ümber olevast elastsest nahast. Ühenduskohtade juures olevatele tippudele tuleb anda "väike" kaalukus (10-40%), et need liiguksid luu liigutamisel vaid pisut. Tippude kaale võib skeleti elustamisel lisada automaatselt (vaata [skelettide teemat](#)), kuid sa võid seda teha ka käsitsi. Selle tegemiseks järgi allolevat protsessi. Automaatselt omistatud kaalude muutmiseks hüppa kirjelduses vajalikku kohta:

- Tekita skelett.
- Loo võre, mida see skelett deformeerima hakkab.
- Lisa sellele võrele töötlejate (Modifiers) saki alt skeletitöötleja (Armature). Sisesta skeleti nimi.

Siit võid jätkata automaatselt loodud kaalude muutmisega.

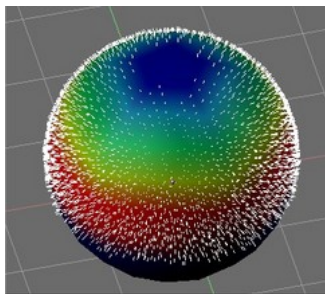
- Märgi 3D-vaates skelett, sisene **poseerimise režiimi Pose**, kasutades klahvikombinatsiooni (Ctrl⇨ Tab, või kasuta 3D-vaate alaservas olevat režiimi valimise menüüd).
- Vali skeletist sobiv luu.
- Vali võre ära ja lülita kaalude maalimise režiimi Weight Paint. Võre on värvitud vastavalt sellele, kui palju vastav luu seda mõjutab. Algseisus on see sinist värvi (mõju puudub).
- Maali tippude kaalud vastavalt sisetundele. Vahetu luu ümbrus peaks olema punane ja kaugemal kergelt vikerkaarevärvides hajuma.

Sa võid mõne teise luu valida parempoolse hiireklahviga klõpsates. Kui võre on luude peal, siis pole neid maalimise ajal näha. Et teha skelett läbi teiste objektide nähtavaks, märgi esmalt objektirežiimis vastav skelett ja siis vali objekti andmete (Object Data) saki alt Display seksioonist X-Ray. Samast kohast saad muuta ka luude kuvamise viise, milleks on oktaeedriline (Octahedron), pulk (Stick), B-luu (B-Bone) või ümbrik (Envelope). Luude nimede kuvamiseks lülita sisse valik Draw Names - siis saad veenduda, et tipugruppide nimed ja luude nimed oleksid samasugused.

Võre maalimisel tekitatakse iga luu kohta üks tipugrupp. Väljaspool tipugrupi ala olevate tippude maalimisel lisatakse need automaatselt tipugruppi.

Sümmeetrilise võre ja skeleti korral võib kasutada peegeldamise (X-Mirror) võimalust. Siis juhtub nii, et ühel pool tehtud maalimine peegeldub ka teisele poolele.

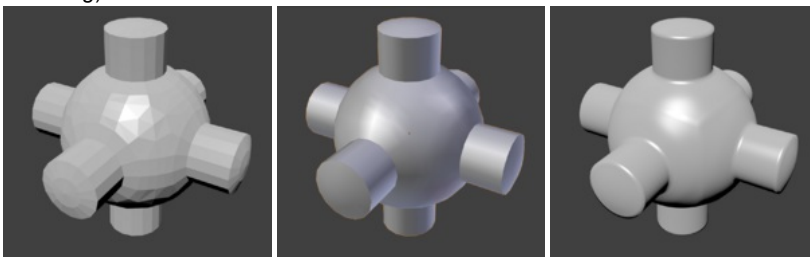
Kaalude maalimine osakeste genereerimiseks



Osakeste genereerimine
tipukaalude alusel.

Nullväärtusega tippudest koosnev külg osakesi ei tekita. Mõju suurus 0.1 tingib osakeste tekkimise kümneprotsendise tihedusega maksimaalsest. Selle abil saab osakeste genereerimise piirkondade vahel ära jagada nii, et suurema kaaluga aladele jääb rohkem osakesi kui väiksema kaaluga aladele. Nii saab näiteks määrata, et mingi osa võrest oleks rohkem või vähem karvane. Osakeste generaatori all on ka muude omaduste määramiseks võimalik tipugruppide kasutamine.

Võre silumine (*Mesh smoothing*)



Võreobjektid, renderdatud lameda varjutusega, silutud servade lõhestamise (*edge split*) ning tükeldamise (*Subdivision Surface*) abil. Vaata, kuidas toimub servade erinev renderdamine. [.blend-näitefail](#)

Nagu näha eelmistest osadest, on polügoonid Blenderis tähtsal kohal. Enamik objekte on esitatud hulknurkadena ja samuti on kõveratest koosnevad objektid sageli esitatud polügonaalse võrena. Renderdamisel võib näha olla, et need koosnevad seeriast pisikestest lamedatest külgedest.

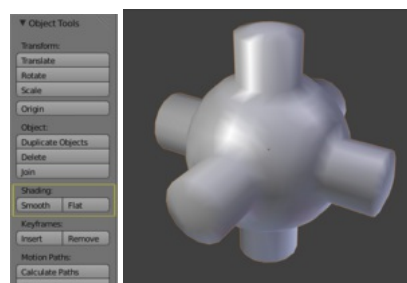
Mõnikord on see soovitud efekt, kuid tavaliselt tahame me, et objektid oleksid sujuvad. Siin osas tuleb juttu sellest, kuidas sama objekti sujuvana näidata ja kuidas kasutada automaatset silumist (Auto Smooth).

Selle osa lõpus tuleb juttu näiva sujuvuse asemel võre tegelikult sujuvaks muutmisest.

Sile varjutamine (*Smooth shading*)

Kõige lihtsam viis objekti siledalt kuvamiseks on lülitada objektirežiimis olles tööriistariist (Tool Shelf) sisse nupp Smooth (silumine). See nupp ei jää visuaalselt sissevajutatuks, sest mõjutab terve võre asemel tegelikult igat üksikut võre külge. Selle nupu vajutamine objektirežiimis tingib selle, et ka edaspidi muutmisrežiimis geomeetria lisamisel on need koheselt siledalt varjutatuks määratud.

Nagu näha, on objekti äärejoon endiselt tugevalt kandiline. Silumine ei muuda tegelikult objekti geomeetria; muutub ainult see, kuidas toimub varjutuse väljaarvutamine pinna siledamana kuvamiseks. Kandilise varjutuse tagasisaamiseks vajuta eelneva nupu kõrval olevat Flat (lame) nuppu.



Sama võre siledalt varjutatuna

Võre osade silumine

Alternatiivse variandina võib siluda ka ainult mõned servad. Selleks mine muutmisrežiimi (Edit mode), tee valituks mõned küljed ja vajuta Smooth (silu) nuppu. Valitud külgede servad kuvatakse kollasena.

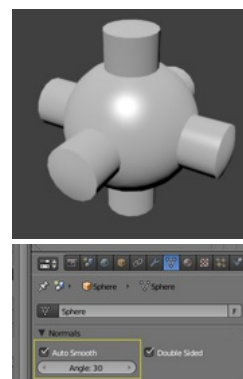
Muutmisrežiimis olles saavad "silutud" omaduse vaid märgitud servad. Samamoodi saab külgi ühekaupa "kandiliseks" muuta. Märgi soovitud küljed ära ja vajuta nuppu Flat (lame).

Automaatne silumine (*Auto Smooth*)

Selle meetodiga üksi on raske lamedate ja silutud pindade kombinatsiooni muuta. Üks meetoditest on külgede eraldamine. Märgi muutmisrežiimis mõned küljed ja vajuta nende võrest lahtiühendamiseks klahvi Y).

Veel üks variant silumiseks on objekti andmete (*Object Data*) saki alt automaatse silumise (*autosmooth*) sisselülitamine. Seal määratud nurgast Angle väiksema nurgaga küljed renderdatakse silutult ja suurema nurgaga mitte. Selle muutmine mõjutab ainult renderdatud pilti - 3D-vaatele see mõju ei avalda. Sinna suurema numbri panemine muudab rohkem silutuks ka teravama servaga küljed.

Tuleb meele pidada, et kui ükskõik millisele küljele on omistatud lame varjutus (Flat), siis automaatse silumise (Auto Smooth) sisselülitamine neid konkreetseid külgi ei mõjuta, need jäävad ikka kandiliseks.



Näidispilt võrest automaatse silumisega (Auto Smooth)

Servalõhestaja töötleja (*Edge Split Modifier*)



Servalõhestaja sisse

lülitatuna

[Servalõhestaja](#) töötleja abil saab automaatse silumise (Auto Smooth) kasutamise sarnase tulemuse, kuid lisaks saab määrata, milliseid servasid lõhestatakse.

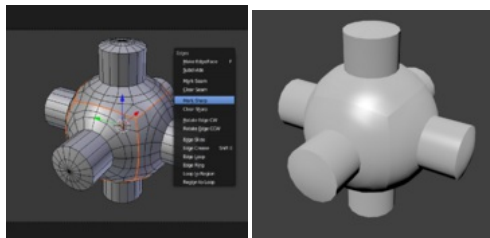
Edge Angle (serva nurk)

Selle abil saab määrata, millise nurgaga servad veel silutakse. See on sama nagu automaatse silumise (Auto Smooth) puhul.

Nurk võib olla vahemikus 0° kuni 180°.

Sharp Angles (teravad nurgad)

Märgi teravaks märgitud servade abil, milliseid nurki lõhestada. Muutmisrežiimis võib Alt+Shift RMB abil servasilmused ära märkida ja klahvikombinatsiooniga CtrlE avanevast menüüst Mark Sharp valides nad teravaks määrata.



Märgitud servasilmuste teravaks määramise abil saame määrata, kuidas võre renderdatakse.

Võre geomeetria silumine

Seninäidatud meetodid muutsid ainult võre kuvamise viisi. Geomeetriat ennast need ei mõjuta. Siledana kuvamise asemel võib sileda tulemuse saamiseks ka võre geomeetriat ennast siluda.

Võre muutmise tööriistad

Muutmisrežiimis võid rakendada järgnevaid:

- [Smooth \(silu\)](#)

See muudab võre sujuvamaks

- [Subdivide Smooth \(tükelda ja silu\)](#)

See jaotab võre väiksemateks segmentideks nii, et vaheliited teevad võre ümaramaks. See sarnaneb tükeldamise töötlejale.

- [Bevel \(\(kandi\)\)](#)

See "viilib" märgitud servad lamedamaks.

Töötlejad (*Modifiers*)

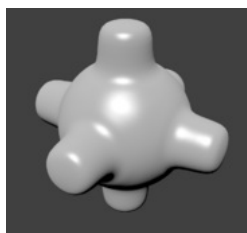
Alternatiivina saab võre sujuvamaks muuta ka seda otseselt mitte muutes, vaid kasutades töötlejaid:

[Silumistöötleja \(*Smooth Modifier*\)](#)

Toimib nagu muutmisrežiimi all olev Smooth. Seda saab rakendada soovitud tippudele tipugruppide abil.

[Kantimistöötleja \(*Bevel Modifier*\)](#)

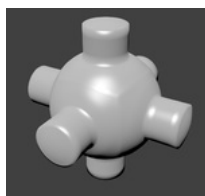
Toimib nagu muutmisrežiimis olev kantimise (Bevel) tööriist; seda saab tööle rakendada vastavalt nurgale või serva kaalu väärtusele.



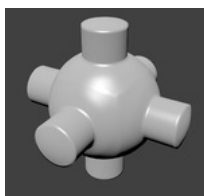
Tükeldaja (*Subsurf*)

[Pinnatükeldaja \(*Subdivision Surface Modifier*\)](#)

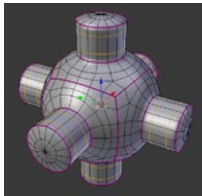
Catmull-Clarki pinnatükeldus annab sujuvama tulemuse. Teravaid servi saab määrata [pinnatükelduse viikide \(*subdivision creases*\)](#) abil või vastavaid servi teravaks märkides ja määraates neile [servalõhestaja töötleja \(*EdgeSplit modifier*\)](#) pärast pinnatükeldaja (Subsurf) töötlejat.



Volditud servade kasutamine ja selle kõrvalmõjud



Täiendavate servasilmuste lisamine



3D-vaates on voldid liilad ja servasilmused kollased

Kõverad (Curves)

Kõverad ja [pinnad](#) (*surfaces*) on objektid nagu võredki, kuid neid esitatakse matemaatiliste funktsioonide, mitte punktide kogumina.

Blender kasutab nii [Bezier'](#) kõveraid kui ka NURBS-kõveraid ja pindasid ([NURBS](#), *Non-Uniform Rational B-Splines* ehk ebaühtlased ratsionaalarvulised Bezier' kõverad). Mõlema variandi puhul defineerib grupp "kontrolltippe" (ehk "kontrollpunkte") "kontrollpolügooni", kuid see defineerimine toimub erinevate matemaatiliste seaduste alusel. Moodus, kuidas kõverat ja pinda interpoleeritakse, võib esmapilgul näida samane Catmull-Clarki tüüpi pinnatükeldusele. Kõver on *interpoleeritud*, kuid pind on sellele moodustatud *pinevusega*.

Võrdlus polügonaalise võrega

Üks peamisi eeliseid, mis kõveratel on polügonaalsete võrede ees, on see, et kõveraid määratletakse väiksema hulga andmetega ning seetõttu võib nendega saavutada väga häid tulemusi, kasutades modelleerimise ajal vähem mälu ja salvestusruumi. Kuid taoline protseduuriline lähenemine pindadele võib renderdamisel nõudmisi hoopis suurendada.

Teatud modelleerimise tehnikad, näiteks rada järgiva profiili [eendamine](#) on võimalikud ainult kõveraid kasutades. Samas on tippude tasemel kuju kontrollimine kõveraid kasutades keerulisem.

On olukordi, kus kõverate ja pindade kasutamine on kasulikum kui võrede kasutamine, kuid samas on ka olukordi, kus võredel on eeliseid. Soovitame sul lugeda läbi [võredega modelleerimise](#) peatüki, mis peaks aitama otsustada, kas projekti puhul tuleks kasutada võresid või kõveraid.

Kui sa oled lõpetanud Bezier' ja NURBS-kõveratest lugemise ja oled need selgeks saanud, siis on keerukate objektide modelleerimise [õppetükkide](#) all mitu veelgi raskemat näidet kõverate kasutamise kohta.



Logo eelvaade.

Näiteks on seal [näide](#) sellest, kuidas luua huvitavat linnusarnast logo. See õppetükk tegeleb enamiku Bezier' kõveratega töötamise aspektidega: kõverate lisamine, taustapildi määramine modelleerimise näidiseks ning saadud kõvera kantimine.

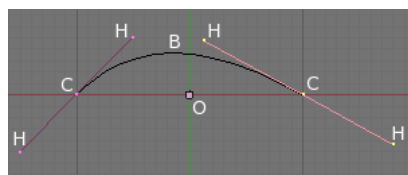
Lisaks on õppetükkidel all veel näidiseid nii [katmise](#) kui ka [kõveratega moonutamise](#) tehnikatest.

Kõvera tööriistade leidmine

Enamik kõveratega seotud tööriistu on grupeeritud kolme paneeli alla: kõver ja pind ([Curve and Surface](#)), kõvera tööriistad ([Curve Tools](#)) ja kõvera tööriistad 1 ([Curve Tools1](#)).

Mitmeid võimalusi leiad ka 3D-vaate päises olevatest valimise (Select) ja kõvera (Curve) menüüdest ning eriomaduste menüüst Specials (W).

Visualiseerimine



Bezier' kõvera näidis.



NURBS-kõvera näidis.

Muutmisrežiimis Edit kuvatakse kõveraid mustade joontena ning tahke varjutuse Solid (või varjutatud Shaded) puhul on kõverad täidetud lameda pinnaga. Kolmemõõtmelistel kõveratel jooksevad mööda kõverat täiendavad "nooled", mis näitavad nii suunda kui ka *suhtelist* liikumiskiirust, mille kõverat rajana kasutavad objektid saavutavad (kui ei ole määratud eraldi kiiruse Speed Ipo).

Iga kontrollpunkt on kuvatud lilla värviga. Bezier' "tipud" on tähistatud roosade joontega ning NURBSide tipud on omavahel seotud kollaste joontega. Nagu võrede puhulgi, on valitud tipud esile tõstetud kollastena (aktiivne element on tähistatud heledama tooniga).

Elementide peitmine

Nagu objekti režiimis Object, saad ka siin valitud peita ning peidetut nähtavaks muuta. See on väga kasulik, kui töötad keeruliste tuhandetest tippudest koosnevate mudelitega ja soovid oma vaateaknaid natuke puhastada.

Peitmiseks kasuta kiirvalikut Ctrl+Alt+H, kõvera tööriistade paneelis (Curve Tools1) olevat peitmise nuppu Hide või menüüvalikut Curve » Show/Hide Control Points » Hide Selected.

Peidetu nähtavaks muutmiseks kasuta kiirvalikut Alt+H, kõvera tööriistade paneelis (Curve Tools1) olevat näitamise nuppu Reveal või sobivat valikut menüüs Curve » Show/Hide Control Points.

Kövera struktuur

U ja V koordinaatidest

U ja V on tavapäraselt kasutatavad nimed, mis tähistavad parameetriliste "koordinaatide" osasid (pane tähele, et nende nimed on samad nagu tekstuuri omadel). Nende kaudu määratakse resolutsioon, NURBSide "sõlmed" jne, sõltuvalt valitud "teljest". Köverate puhul on oluline ainult U-osa (sest nad on jooned ja seega on neil ainult üks mõõde). V osa kasutavad NURBS-pinnad.

Kontrollpunktid ja segmendid

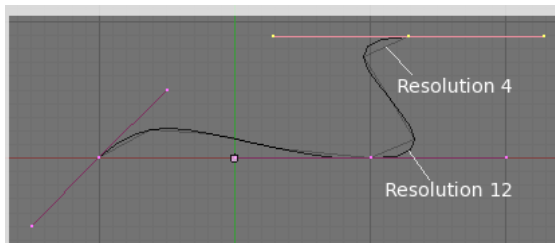
Köverate kuju on määratud kontrollpunktidega ning nendest räägitakse vastavates osades allpool. Kuid mõnikord näed sa ka mõisted kövera "segment". Segmenti määravad kaudselt kaks üksteisega kõrvuti olevat kontrollpunkti. Erinevalt võredest ei saa sa segmenti otse valida, vaid pead selleks valima segmenti määravad kaks kontrollpunkti.

Kövera resolutsioon

Kuigi köverad on pidevad (ühtlaselt jätkuvad) matemaatilised objektid, tuleb neid siiski renderdamise seisukohalt näidata mittepidevas vormis (väikeste segmentide grupina). Seda tehakse läbi kövera *resolutsiooni* sätte, mis määrab punktide arvu, mida kahe kontrollpunkti vahel välja arvutatakse. Pea meeles, et see kehtib ainult Bezier' köverate puhul ning toimib NURBSide puhul teisiti.

Tegelikult on sul kolm resolutsiooni sätet: kaks kövera ja pinna paneelis Curve and Surface ja üks kövera tööriistade paneelis Curve Tools:

- DefResolU (vaikeresolutsioon): vaikeresolutsioon, mis määratakse nii 3D-vaadetes kui ka renderdamise ajal kõigile objekti köveratele.
- RenResolU (renderdamise resolutsioon): resolutsioon, mis määratakse renderdamise ajal kõigile objekti köveratele, juhul kui selle välja väärtus ei ole null (muidu kasutatakse DefResolU väärtust).
- Resol U (resolutsioon): seda sätet saab muuta ainult muutmisrežiimis ning see muudab parajasti (osaliselt) valitud köverate resolutsiooni. Pane tähele, et pärast selle sätte igat muutmist määratakse selle väärtuseks uuesti DefResolU väärtus.



Bezier' resolutsiooni näide.

Igale köverale saab määrata eraldi *resolutsiooni*, muutes [kövera ja pinna paneelis Curve and Surface](#) oleva vaikeresolutsiooni välja DefResolU väärtust. Vaikerväärtus on **6**. Pilt (*Bezier' resolutsiooni näide*) näitab sama köverat, mis on pilditöötlusprogrammis Gimp asetatud erinevatel resolutsiooni sätetel iseendaga kohakuti. Heledama kövera resolutsioon on madal **4** ning see köver hakkab välja nägema nurgeline. Tumedama kövera resolutsioon on **12** ja on seetõttu väga sujuv. Pea meeles, et kõrgem resolutsioon võib näha välja parem, kuid suure hulga köverate puhul (ja/või siis, kui suurel hulgal köveratel on palju kontrollpunkte) võib see aeglustada interaktiivset renderdamist.

Kahe- ja kolmemõõtmelised köverad

Vaikimisi (välja arvatud raja primitiivi Path puhul) on uued köverad kahemõõtmelised ehk kõik nende kontrollpunktid asetsevad samal tasapinnal (X-Y kohalikus ruumis). Seeläbi saab suletud köveraid automaatselt täita.

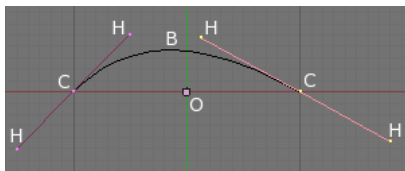
Kuid sa võid soovida kontrollpunkte vabalt kolmemõõtmelisse ruumi paigutada. Selle saavutamiseks lülita sisse nupp 3D objekti andmete paneelis Object Data. Pane tähele, et sa kaotad seeläbi suletud köverate täite ning köveral näidatakse nooli nii, nagu seletati eelnevas [visualiseerimise](#) osas.

Suletud ja avatud köverad

Köverad (nii Bezier' köverad kui ka NURBSid) on ühemõõtmelised: nad on jooned, mis ei ristu kunagi iseendaga (st kõik kontrollpunktid on ühenduses maksimaalselt kahe teise kontrollpunktiga ning köver moodustab katkematu joone, ilma et hargneks). Kuid köverad võivad olla avatud (st algus- ja lõpp-punkt pole omavahel kövera segmendiga ühendatud) või suletud. Suletud köverad on tahked kehad, sest Blender lisab nende sisse tahke lameda külje (juhuil kui nad pole tehtud kolmemõõtmelisteks).

Bezier' köverad

Bezier' köverad on kõige tavapärasemad köverad, mida kasutatakse tähtede ja logode disainimiseks. Neid kasutatakse laialdaselt ka animatsioonis - nii radadena, mida mõõda objektid saavad liikuda, kui ka IPO-köveratena, mis muudavad aja funktsioonina objektide omadusi.



Näide köverast.

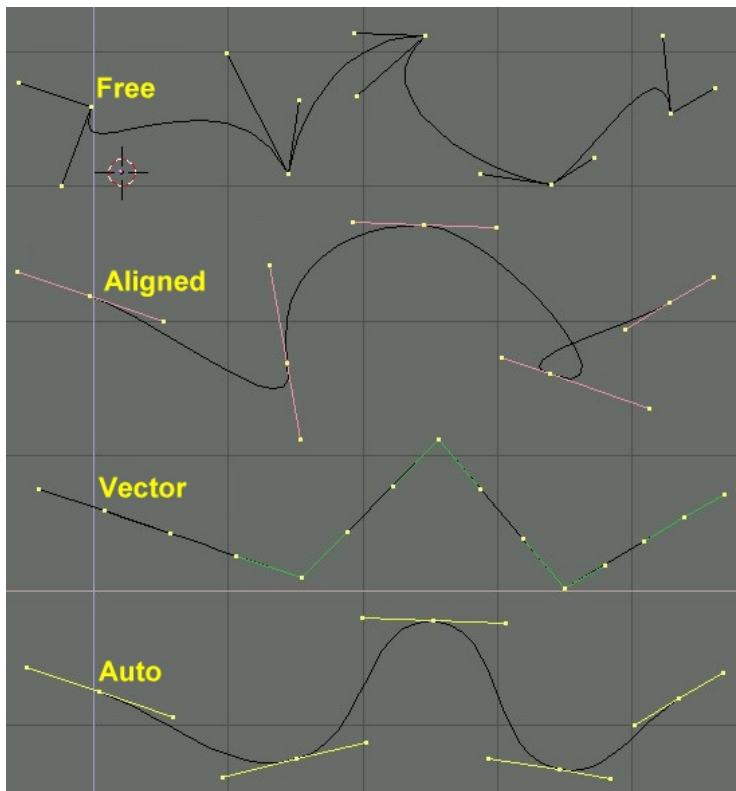
Pilt (*Näide köverast*) kujutab kõige lihtsamat Bezier' köverat, mida on võimalik luua. See koosneb kahest kontrollpunktist ehk tipust,

mida tähistab "c", kõverast "b", sangadest "h" ning objekti keskmest "o".

Sang "h" määrab punktis "c" oleva kõvera tangensvektori (vastaval küljel). Kõvera "järskus" saab määrata läbi sanga pikkuse ("h" vastavale punktile "c"). Mida pikem on sang, seda järsem on kõver (st seda enam soovib kõver sanga *kallistada*).

Sangu on nelja tüüpi (*Bezier' kõverate sangade tüübid*):

- Vaba sang (Free, must) Sangad on üksteisest sõltumatud. Et muuta sang vabaks, kasuta kiirvalikut H. Valik H lülitab samuti vaba (Free) ja joondatud (Aligned) sangatüübi vahel.
- Joondatud sang (Aligned, lilla). Need sangad on alati teineteise suunaliselt paigutatud (ja vastassuunalised) ning tulemuseks on püsiv kõver (ilma teravate nurkadeta). Kiirvalik: H (lülitab vaba (Free) ja joondatud (Aligned) sangatüübi vahel).
- Vektorsang (Vector, roheline). Sanga mõlemad osad on alati suunatud kas eelmise või järgmise sanga suunas (seega saad luua sirgetest joontest koosnevaid kõveraid!). Kiirvalik: V.
- Automaatne sang (Auto, kollane). Sel sangal on täiesti automaatne pikkus ja suund, mille määrab Blender, et saavutada sujuvat tulemust. Kiirvalik: \oplus ShiftH.



Bezier' kõverate sangade tüübid.

Sangu saab *liigutada*, *pöörata* ja *suurendada-vähendada* nagu tavalisi võre tippegi. Niipea kui *sanga* liigutatakse, muudetakse automaatselt sanga tüüpi:

- Automaatne sang (Auto) muutub joondatud sangaks (Aligned);
- Vektorsang (Vector) muutub vabaks sangaks (Free).

NURBS-kõverad

NURBS-kõverad on defineeritud ratsionaalarvuliste polünoomidena ning on lihtsalt öeldes üldisemad kui tavapärased [Bezier' kõverad](#), sest nad on suutelised järgima täpselt ükskõik millist kontuuri. Näiteks Bezier' ring on ringi polünoomne *lähend* ning taoline lähendus on märgatav, kuid NURBS-ring on *täpne*. Kuid NURBSid ei saa erinevalt Bezier' kõveratest sisaldada tõeliseid teravaid nurki.

NURBS-kõverate puhul tuleb selleks, et neid täiel määral ära kasutada, mõista natuke enam neid loovate osade käitumist. Nad koosnevad suurest hulgast muutujatest, mille abil saad luua matemaatilisel puhtaid vorme. Kuid selleks, et nendega töötada, peame natuke rääkima erinevatest NURBS-kõvera osadest.

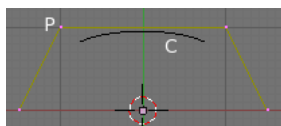
Märkus

NURBSid on *väga* keeruline teema; ma arvan, et nad oleks väga hea teema ühele "teooriat" käsitlevale artiklile – mille kirjutaks keegi, kes neid sügavuti mõistab! (Alustuseks on olemas [inglise keelse Wikipedia lehekülje](#).)

Sõlmed (*Knots*)

Alustame *sõlmedega*. NURBS-kõveratel on *sõlme* tipp, mis kujutab endast numbrite rida, mis määrab kõvera parameetrid (st nad kirjeldavad iga kontrollpunkti mõju ulatust). Tuleta meelde Bezier' kõvera kontrollpunkte - NURBSidel on need samuti olemas ning iga kontrollpunkt mõjutab oma mõjualas mõnda kõvera osa. Kontrollpunkte kuvatakse lillade tippudena.

Pea meeles, et sõlmede säte puudutab ainult *avatud* NURBS-e, mitte suletuid.

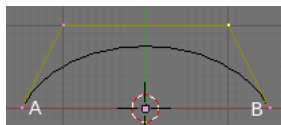


Ühtlane vaikekõver.

Pildil (*Ühtlane vaikekõver*) on NURBS-vaikekõver, mis luuakse NURBS-kõvera menüüvalikuga NURBS Curve tööriista kasti lisamise menüüs [Add](#) ning see on näide ühtlasest kõverast (Uniform). Kõver ise kuvatakse mustana (tähistatud "c") ning kontrollpunktid lilladena (üks neljast on tähistatud "P").

Sõlme tippu sa otseselt muuta ei saa, kuid saad selle kolme eelvaliku kaudu määratleda: ühtlane (Uniform), lõpp-punkt (Endpoint) ja Bezier (Bezier).

Ühtlase nupp Uniform kõvera tööriistade paneelis [Curve Tools](#) loob suletud kõverate puhul ühtlase jagunemise, kuid avatud kõverate puhul tekivad "vabad" otsad, mida on keeruline üles leida.

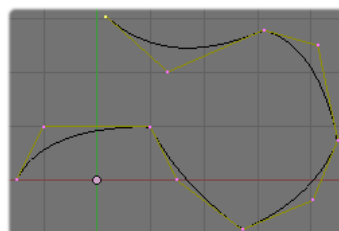


Lõpp-punkti kõver.

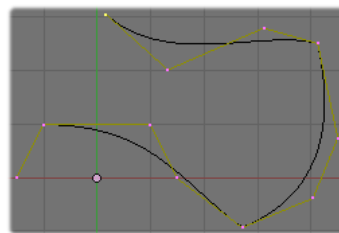
Lõpp-punkti nupp Endpoint määrab *sõlme* tipu sedasi, et esimene ja viimane tipp on alati kõvera osad, mis läbi on nende asetamine lihtsam. Pilt (*Lõpp-punkti kõver*) näitab, mis juhtub siis, kui kasutad pildil (*Ühtlane vaikekõver*) toodud kõvera puhul nuppu Endpoint. Näed, et kõver on nüüd tõmmatud *lõpu* kontrollpunktideni, mida tähistavad "A" ja "B".

Ja Bezier' "*sõlme tipp*" Bezier (ma ei ole täiesti kindel, kas see säilib reaalse sõlmena) paneb NURBS-i kontrollpunktid jäljendama Bezier' kontrollpunktide käitumist. See töötab ainult kahe järgu väärtusega (loe allpoolt): **3** ja **4**.

- Järgu Order väärtuse **3** puhul käituvad kõik paaritud punktid Bezier' kontrollpunkti keskohtadena ja paaris punktid Bezier' sangadena. Seega määravad tipud (1,2,3) esimese Bezier' segmendi, (3,4,5) teise ja nii edasi.
- Järgu Order väärtuse **4** puhul määravad kõik neli kontrollpunktide gruppi Bezier' segmendi; st tipud (2,3,4,5) määravad esimese segmendi, (5,6,7,8) teise, jne. Pane tähele, et sel puhul esimest tippu millegipärast ei kasutata.



Kolmandat järku Bezier' NURBS.



Neljandat järku Bezier' NURBS.

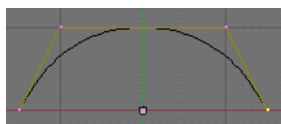
Järk

Järgu väli Order kõvera tööriistade paneelis [Curve Tools](#) tähistab kõvera "sügavust" või *kraadi* (st sa määrad, mitu kontrollpunkti võetakse kõvera kuju arvutamisel arvesse).

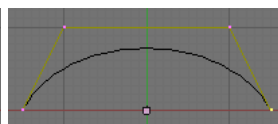
Order **1** on punkt ning seetõttu pole sügavuse sätena kasutatav, Order **2** on lineaarne (*Teise järku kõver*), Order **3** on ruutfunktsiooni põhine (*Kolmanda järku kõver*), (*Neljanda järku kõver*) on kuupfunktsioon ja nii edasi. Sobivad väärtused on vahemikus **2** kuni **6**. Pane tähele, et järgu Order kasvuga liigub kõver kontrollpunktidest eemale.



Teise järku kõver.



Kolmanda järku kõver.



Neljanda järku kõver.

Kui su kõveral on **6** või enam kontrollpunkti, ei saa järku Order määrata kõrgemaks kui **6**. **6** on kõrgeim lubatud järk Order. Kui sul on vähem kui **6** kontrollpunkti, siis on järgu Order väärtus piiratud kontrollpunktide arvuga. Kui su kõveral on näiteks **5** kontrollpunkti, siis suurim järgu (Order) lubatud väärtus on **5**.

Kasuta võimaluse korral alati järgu väärtust **5**, sest see toimib kõigis tingimustes voolavalt ega loo liikumises häirivaid ebaühtlusi. Kui sa paned kuubi liikuma mööda NURBS-i rada, mille järk on näiteks **2**, siis liigub kuup mööda rada ebasujuvalt (nõksutades).

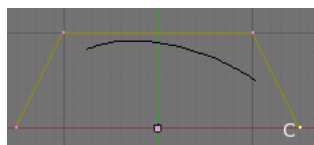
Märkus matemaatikast

Matemaatiliselt öeldes tähistab järk (Order) nii NURBS-kõverat määrava ratsionaalarvulise polünoomi lugejat kui ka nimetaja astendajat.

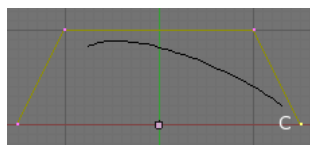
Kaal

NURBS-kõveratel on kaal Weight (kõvera tööriistade paneelis [Curve Tools](#)), mis on määratud igale kontrollpunktile ja tähistab seda, kui palju iga punkt kõverat "tõmbab". Kujuta ette, et igal kontrollpunktil on käsi, mis sirutab ennast välja, haarab kõverast kinni ja

proovib seda tirida. Mida suurem on kaal (Weight), seda rohkem kontrollpunkt kõverat tõmbab; vaata pilte (*Kaal 5*) ja (*Kaal 20*). Kaalu sätte (Weight) väärtuste lubatud vahemik on **0.1** kuni **100.0**.

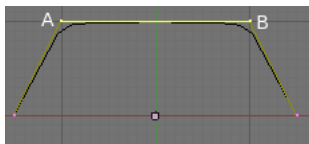


Kaal 5.



Kaal 20.

Suurem kaalu väärtus 20 tõmbab kõverat tähistusega "c" kontrollpunkti suunas. Igal kontrollpunktil saab olla erinev kaalu sätet. Kui kontrollpunkti kaal suureneb, liigub kõver kontrollpunktile lähemale. Kui kaalud on piisavalt suured, järgib kõver kontrollpunkte peaaegu täpselt; vaata pilti (*Kaal 100*).



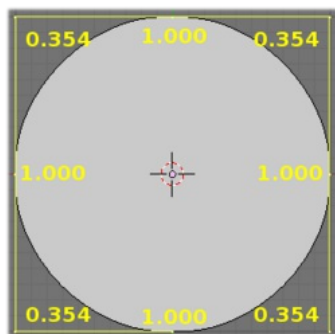
Kaal 100.

Kontrollpunktid saavad nõnda omavahel võistelda. Näiteks kõige suurema kaaluga kontrollpunkt tõmbab kõverat enda poole ja teistest eemale. Kui kõigil kontrollpunktidel on sama kaal, siis põhimõtteliselt raskuse väärtused tühistatakse, justkui puuduksid raskused täielikult.

Pildil (*Kaal 100*) on kahe ülemise kontrollpunkti kaalu väärtus **100.0** (tähistatud "A" ja "B"). Vastasseisvate kontrollpunktide kaalu väärtus on **1.0**. Näed, kuidas kõverat "tõmmatakse" kontrollpunktide "A" ja "B" suunas. Ja sedavõrd suure kaalu väärtuse puhul järgib kõver peaaegu täpselt kontrollpunkte.

Nägemaks kontrollpunkti kaalu väärtust, vali see, ava teisendussätete paneel [Transform Properties](#), kasutades kiirvalikut N, ning vaata tipukaalu välja Vertex W. Kaalu väli Weight ei näita kaalu ennast!

Eelseadistatud kaalud



Ringikujuline NURBS-kõver.

NURBS-kõveratega saab luua puhtaid kujundeid nagu ringe (pea meeles, et Bezier' ringid ei ole puhtad ringid). Puhaste ringide loomiseks pead määrama kontrollpunktidele kindlad kaalud, millest mõned on eelseadistustena kõvera tööriistade paneelis Curve Tools ära toodud (alumine parempoolne nurk). See ei ole kohe mõistetav ning sa peaksid enne nende proovimist lugema NURBSide kohta täpsemalt.

Põhimõtteliselt peab ringikujulise kaare loomiseks kolme kontrollpunktiga kõverast olema lõpp-punktidel sama kaal ning keskmise kontrollpunkti kaal peab olema pool punkte ühendavate segmentide vahelise nurga koosinusest. Vaatame näidet: kui sul on kolm kontrollpunkti, mis moodustavad täisnurga, siis pead sa kvadranti tekitamiseks määrama keskmise punkti kaaluks $\cos(90/2)/2 = \cos(45)/2 = \text{ruutjuur}(2)/4 = 2^{*-1.5} = 0.354$ (need sätet töötavad järgu väärtuse **4** puhul, seega peab sul olema vähemalt neli kontrollpunkti, sest kolme punktiga – ja seega järgu väärtusega kolm **3** – ei saavuta sa "ringikujulist" kvadranti!).

Teised sätet on kasulikud [NURBS-pindade puhul](#) (loomaks keraid, silindreid jms).

Primitiivid (*Primitives*)

Blenderis saad lisada viit erinevat kõvera primitiivi, kaks Bezier' ja kolm NURBSi oma:

- Bezier' kõver (Bezier Curve) lisab kahe kontrollpunktiga avatud kahemõõtmelise Bezier' kõvera.
- Bezier' ring (Bezier Circle) lisab suletud ringikujulise kahemõõtmelise Bezier' kõvera (mis koosneb neljast kontrollpunktist).
- NURBS-kõver (NURBS Curve) lisab avatud kahemõõtmelise NURBS-kõvera, millel on neli kontrollpunkti ja ühtlased sõlmed (Uniform).
- NURBS-ring (NURBS Circle) lisab suletud ringikujulise kahemõõtmelise NURBS-kõvera (mis koosneb kõrguse kontrollpunktidest).
- Rada (Path) lisab avatud kolmemõõtmelise NURBS-kõvera, mis koosneb viiest kontrollpunktist, millel on lõpp-punkti sõlmed Endpoint ning mille kõvera-aja sätet CurvePath on sisse lülitatud.

Köverate muutmine

Köverate muutmisel on vähem tööriistu ja võimalusi kui võre muutmisel. See ei tähenda aga, et see oleks lihtsam või vähem võimas!

See lehekülg räägib algtasemel köverate muutmisest – keerulisematest teemadest nagu eendamine (ja kantimine ja koondamine) räägitakse järgmistes peatükkides: [Köveraga moonutamine](#) ja [Eendamine](#).

Lihtne köverate muutmine (liigutamine, pööramine, suuruse muutmine)

Hotkey: G/R/S

Menu: Curve » Transform » Grab/Move, Rotate, Scale, ...

Kui sa oled ühe või rohkem kontrollpunkte valinud, saad neid haarata/liigutada (G), pöörata (R) või muuta nende suurust (S) nagu paljude teistegi asjade puhul Blenderis ja nagu seda on kirjeldatud [3D-ruumis manipuleerimise](#) peatükis.

Pane tähele, et Bezier' köverate kontrollpunktid koosnevad kolmest tipust (keskmine ja kaks sanga) ning seega on tervenisti valitud kontrollpunkt teisendamise tööriistade jaoks võrdväärne kolme valitud tipuga (st erinevalt tavalisest võrest ja NURBS-tippudest saad seda pöörata ja selle suurust muuta).

Samuti pea meeles, et üldiselt on Bezier' köveraid lihtsam muuta kui NURBS-köveraid, sest muutes kontrollpunkti, muudad sa vaid sellest punktist teisele poole jäävaid köverate segmente. NURBSide puhul võib ühe tipu liigutamine muuta kuni kolme punkti mõlemal poolel, sõltuvalt kõvera järgust. Lisaks jookseb Bezier' kõver läbi kõigi oma kontrollpunktide keskkoha - NURBSid pole sugugi samavõrd lihtsad!

Muutmisrežiimis Edit on sul nende lihtsate muutmiste puhul olemas veel üks lisavõimalus: [proportsionaalne muutmine](#).

Keerulisemad teisendamise tööriistad

Menu: Curve » Transform

Teisendamise tööriistu 'kerakujuliseks' To Sphere, 'aja kiiva' Shear, 'mässi' Wrap ja 'lükka/tõmba' Push/Pull on kirjeldatud [3D-ruumis manipuleerimise](#) peatükis.

Kaks teist tööriista, kalluta Tilt ja vähenda/suurenda raadiust Shrink/Fatten Radius, on seotud [kövera eendamisega](#).

Bezier' kontrollpunktide sätted

Panel: Teisendamise omadused Transform Properties (N)

Hotkey: ⇧ ShiftH, H, V

Menu: Curve » Control Points » Automatic, Toggle Free/Aligned, Vector

See on oluline ainult Bezier' köverate puhul. Nagu me nägime [eelmisel lehel](#), on sellistel köveratel nelja tüüpi sangu (mis annavad sujuva või nurgelise kõvera). ⇧ ShiftH muudab kõik valitud kontrollpunktid automaatseteks, H lülitab vabade ja joondatud vahel ning V muudab nad vektor-tüüpi punktideks. Loe täpsemalt ülaloleva lingi alt.

NURBSide kontrollpunktide sätted


Panel: Kövera tööriistad Curve Tools (muutmise kontekst Editing, F9), ja teisendamise omadused Transform Properties


Samuti nägime [eelmisel lehel](#), et NURBSide kontrollpunktidel on olemas raskus, mis määrab nende mõju kõverale. Saad seda määrata kas kõvera tööriistade paneelis Curve Tools asuva suure raskuse määramise nupu Set Weight abil (olles kõigepealt määranud paremale jäävas numbriväljas raskuse) või trükkides väärtuse otse teisendamise omaduste paneelis Transform Properties olevasse numbrivälja W.

Uute segmentide lisamine

Hotkey: Ctrl LMB  või E

Menu: Curve » Extrude

Kui kõver on loodud, saad sellele uusi segmente lisada (tegelikult uusi kontrollpunkte, mis defineerivad segmente), kas köverat eendades või lisades kombinatsiooniga Ctrl LMB  uusi sangasid. Iga uus segment lisatakse ühte kõvera otsa. Uus segment lisatakse ainult siis, kui ühes kõvera otsas on valitud üksik tipp või sang. Kui valitud on kaks või enam kontrollpunkti, ei lisata midagi (kui sa aga kasutasid eendamise käsku E, muudetakse kõigi valitud kontrollpunktide režiim haaramiseks (Grab)).

Pea meeles, et erinevalt võredest ei saa sa muudetud objekti sisse luua uut köverat ilma midagi valimata, lihtsalt kombinatsiooniga Ctrl LMB  vajutades – selleks pead sa kõigepealt olemasoleva kõvera kaheks lõikama, ([kustutades ühe segmendi](#)), [kopeerima](#) mõnda olemasolevat (⇧ ShiftD) või lisama uue (lisamise menüü Add).

Köverate sulgemine ja avamine

Hotkey: AltC

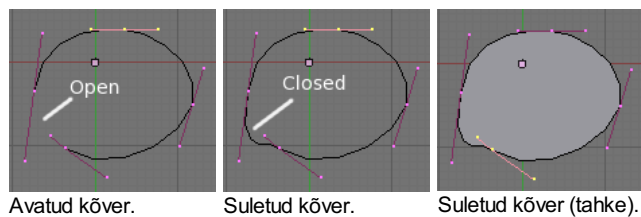
Menu: Curve » Toggle Cyclic

Niimoodi saad vahetada avatud ja suletud köverate vahel. Ainult köverad, millel on valitud vähemalt üks kontrollpunkt, suletakse/avatakse.

Sulgeva segmendi kuju oleneb Bezier' köverate puhul algus- ja lõpp-sangadest ning NURBSide puhul, nagu tavaliselt, naabruses olevatest kontrollpunktidest. Sanga muudetakse sulgemisel ainult siis, kui see on automaatset tüüpi (Auto). Piltidel (*Avatud kõver*) ja (*Suletud kõver*) on näidatud sama Bezier' köverat avatu ja suletuna.

Seda toimingut on võimalik läbi viia ainult algse alguspunkti või viimati lisatud kontrollpunktiga. Segmendi/segmentide kustutamine ei mõjuta toimimise nõudeid, seda saab ikkagi teha ainult alguse ja viimase kontrollpunktiga. See tähendab, et valik AltC võib tegelikult ühendada kaks kõverat, selle asemel et ühtainsat sulgeda!

Pea meeles, et suletud kõver tekitab renderdatava lameda pinna.



Duplitseerimine (*Duplication*)

Hotkey: ⇧ ShiftD

Menu: Curve » Duplicate

See käsk lihtsalt duplitseerib valitud kontrollpunktid ning kaudselt valitud kõvera segmendid (kui neid on). Koopia valitakse ning see on haaramisrežiimis, nii et saad selle teise kohta liigutada.

Elementide kustutamine

Hotkey: X või Del

Menu: Curve » Delete...

Kõverate kustutamise hüpikmenüüs Erase on kolm valikut:

Selected (valitud)

See kustutab valitud kontrollpunktid *ilma* kõverat katki tegemata (st naabruses olevad punktid ühendatakse omavahel kohe pärast vahepealsete kustutamist). Pea meeles, et NURBSide järk ei saa olla kõrgem kui kontrollpunktide arv ning seega võib see pärast kontrollpunktide kustutamist väheneda. Loomulikult, kui jääb ainult üks punkt, ei eksisteeri enam nähtavat kõverat, ning kui kustutatakse kõik punktid, kustutatakse ka kõver ise.

Segment

See valik on mingil määral vastupidine eelmisele, sest see lõikab kõvera katki, eemaldamata ühtegi kontrollpunkti, kuid kustutades ühe valitud segmendi.

See valik eemaldab alati *ainult ühe segmendi* (viimati "valitu"), isegi siis, kui valikus on neid mitu. Seega pead sa kõigi valikus olevate segmentide kustutamiseks kasutama korduvalt sama võimalust.

All (kõik)

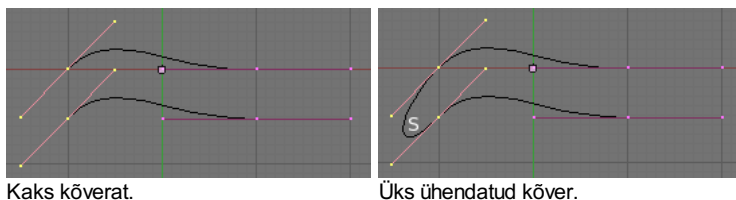
Nagu võrede puhulgi, kustutab see kõik objektis sisalduva!

Kõverate ühendamine

Hotkey: F

Menu: Curve » Make Segment

Kõverate ühendamine tähendab tegelikult kahe kõvera vahele segmendi loomist. Kahe eraldi kõvera ühendamiseks kasuta mõlemast kõverast ühte lõpp-kontrollpunkti. Kaks kõverat ühendatakse uue segmendiga üheks kõveraks. Pilt (*Üks ühendatud kõver*) on pildi (*Kaks kõverat*) ühendamise tulemus. Segment nimega "s" on uus kahte kõverat ühendav segment. Kõvera segmentide loomiseks kasuta kiirvalikut F. Sama valik loob võrede puhul servi (ja külgi), seega...



Kõverat ei saa kõverate ühendamise teel sulgeda. Selleks pead sa kõvera [sulgema](#). Saad veateate *Can't make segment* ("Ei saa segmenti teha"), kui proovid ühendada alustavat ja viimast kontrollpunkti kasutades. Näiteks pildil (*Üks ühendatud kõver*) pead sa kõvera sulgemiseks kasutama [sulgemist](#). Sama veateate saad ka siis, kui oled valinud mitte-lõpp-punktid, liiga palju lõpp-punkte jne.

Pea meeles, et saad ühendada ainult sama tüüpi kõveraid (st Bezier'sid Bezier'dega, NURBSse NURBSidega jne)!

Tükeldamine

Panel: Kõvera tööriistad 1 Curve Tools1 (muutmise kontekst Editing, F9)

Hotkey: W » 1 NumPad

Menu: Curve » Segments » Subdivide, Specials » Subdivide

Kõvera tükeldamine on väga lihtne: kasutades kas tükeldamise valikut Subdivide eriomaduste menüüs Specials (W) või kõvera tööriistade 1 paneelis Curve Tools1 olevat tükeldamise nuppu Subdivide, tükeldad sa igat valitud segmenti ühe korra, lisades täpselt iga valitud kontrollpunktide paari vahele ühe uue.

Muuda suunda (*Switch Direction*)

Hotkey: W » 2 NumPad

Menu: Curve » Segments » Switch Direction, Specials » Switch Direction

See käsk pöörab ükskõik millise kõvera, mille puhul on valitud vähemalt üks element, suuna "tagurpidi" (st alguspunkt lõpetab ja *vastupidi*). See on kasulik, kui kasutad kõverat rajana või koos kantimise ja koondamise sätetega.

Teised eriomadused

Hotkey: W

Menu: Specials

Eriomaduste menüüs Specials on mõned lisavõimalused:

Set Goal Weight (määra kaalu eesmärk)

See määrab valitud kontrollpunktide "kaalu eesmärgi", mis on leitav ka teisendamise omaduste paneeli Transform Properties kaalu sätte Weight alt. Kaalu eesmärk muudab kontrollpunkti tõmbe tugevust. Mida suurem kaal, seda lähemalt möödub kõver kontrollpunktist.

Set Radius (määra raadius)

Seda sätet kasutab [eendamise tööriist](#).

Smooth (silu)

See käsk mõjub samamoodi nagu [võrede](#) puhul, liigutades natuke valitud punkte, et kõverat üldiselt siluda.

Smooth Radius (silu raadiust)

See lihtsalt silub kõvera kontrollpunktide raadiuse väärtust Radius.

Tüübi muutmine

Siinkohal räägime "sisemisest" tüübi muutmisest Bezier' ja NURBSi kõverate vahel.

Eksisteerib ka "väline" muutmine, kõverast võreks, mis töötab ainult objektirežiimis Object. See teisendab kõveraobjekti Curve võreks Mesh, kasutades servade ja tippude loomiseks kõvera resolutsiooni. Pane tähele, et see säilitab ka suletud ja eendatud kõverate poolt loodud mahud ja küljed.

Tagasi "sisemise" muutmise juurde. Kõvera tööriistade paneeli Curve Tools muutmise grupis Convert olevad Bezier' ja NURBSi (Nurb) nupud võimaldavad sul (osaliselt) muuta valitud kõveraid vastavasse tüüpi. Pea meeles, et see ei ole "tark" muutmine, st Blender ei proovi hoida sama kuju ega sama kontrollpunktide arvu. Kui sa näiteks muudad NURBSi Bezier'ks, muutub iga kolmeline NURBSi kontrollpunktide grupp üheks Bezier' omaks (keskpunkt ja kaks sanga).

Mitmesugune muutmine

Sul on ka mõned samasugused valikud nagu võrede puhul või objektirežiimis Object olles. Sa saad valitud kõvera [osadeks jagada](#) (P), muuta teisi valitud objekte ühe või kolme kontrollpunkti [alamateks](#) (CtrlP – pane tähele, et kolmele kontrollpunktile alamaks muutmisel on kõverate puhul imelik tulemus) või [lisada haake](#), et juhtida kõvera punkte teiste objektide abil.

Samuti on kasutatav peegeldamise tööriist Mirror, mis töötab samamoodi nagu [võre tippude](#) puhul, ning [nakkumise](#) käsk.

Topoloogia muutmine

Panel: Kõvera tööriistad 1 Curve Tools1 (muutmise kontekst Editing, F9)



[Võrega nakkumine](#) töötab ka kõvera osade puhul.

See töötab täpselt samamoodi, liigutades valitud kontrollpunkte risti parajasti aktiivse 3D-vaatega, kuni nad pörkuvad vastavalt valikule mõne teise objekti pinna, serva või tipuga (nakkumise töötamiseks pead olema kas tahkes (Solid) või varjutatud (Shaded) kuvamisrežiimis).

Tuleta lisaks meelde, et kahemõõtmeliste kõverate kontrollpunktid on lukustatud samale XY kohalikule tasapinnale ning seega ei saa nakkumine neid mööda kohalikku Z-telge liigutada.

Kõvera valimine

Kõvera valimine muutmisrežiimis Edit on kõvasti lihtsam kui võrede puhul! Peamiselt seetõttu, et saad valida ainult ühte tüüpi elemente - kontrollpunkte (valimisrežiimi ei eksisteeri). Need punktid on aga natuke keerulisemad kui tavalised tipud ning seda eriti Bezier' kõverate puhul, sest neil on nii keskne tipp kui ka kaks selle sanga.

Lihtsad tööriistad on samad mis [võrede](#) puhul, seega saad valida kontrollpunkti lihtsalt klahviga LMB , lisada valikusse klahviga ⇧ Shift LMB , valida piirdkastiga B ja nii edasi.

Üks sõna Bezier' kontrollpunktide kohta: kui valid peamise keskmise tipu, valitakse automaatselt ka mõlemad sangad ning seetõttu

saad seda kõike koos liigutada, ilma et tekitaksid kõverasse nurka. Kui sa aga valid ühe sanga, valitakse ainult sang ise ning saad seeläbi muuta kõverat kontrollivat vektorit.

L (või CtrlL) lisab valikusse hiirekursorile lähima kontrollpunkti ning kõik sellega lingitud punktid, st kõik samasse kõverasse kuuluvad punktid. Pane tähele, et Bezier' puhul tähendab L vajutamine koos valitud sangaga, et valitakse terve kontrollpunkt ja kõik sellega lingitud punktid.

Valiku menüü

Kõverate puhul on kõik "keerulisemad" valikuvõimalused grupeeritud 3D-vaate päises olevasse valikumenüüsse Select. Loeme need üles.

Random... (juhuslik)

Inverse (vastupidine valik)

Select/Deselect All (vali kõik/tühista valik)

Border Select (vali piirdkastiga)

Kõigil neil on sama tähendus ja käitumine kui [objektirežimis Object](#) (ning piirdkastiga valimise erinevustest muutmisrežimis

Edit on juba räägitud [siin](#)).

Iga n. (Every Nth)

Hotkey: Puudub

Menu: Select » Every Nth

See töötab, kui sul on juba valitud vähemalt üks kontrollpunkt. Kasutades parajasti valitud punkti, lisab see valikusse iga n-inda nii enne kui ka pärast parajasti valitud. "Valiku sammu" määrab hüpikväli N, mis ilmub tööriista käivitamisel.

Vali/Tühista valik esimene/viimane (Select/Deselect First/Last)

Hotkey: Puudub

Menu: Select » Select/Deselect First, Select » Select/Deselect Last

Need käsud muudavad objektis oleva(te) kõvera(te) esimes(t)e või viimas(t)e kontrollpunkti(de) valikut. Kasuta seda, et kiiresti leida kõvera algus (nt kui kasutad seda rajana).

Vali järgmine/eelmine (Select Next/Previous)

Hotkey: Puudub

Menu: Select » Select Next, Select » Select Previous

Need käsud valivad vastavalt praegusele valikule järgmise(d) või eelmise(d) kontrollpunkti(d) (st valitutele kõveral järgnevad või eelnevad kontrollpunktid).

Rohkem ja vähem (More/Less)

Hotkey: Ctrl+ NumPad/Ctrl- NumPad

Menu: Select » More/Less

Need kaks valikut vastavad [võre omadele](#) ning käituvad samaselt. Nende eesmärk, sõltuvalt parajasti valitud kontrollpunktidest, on valikut suurendada või vähendada.

Algoritm on sama mis võrede puhul, aga seda on lihtsam mõista:

- More (rohkem): iga kontrollpunkti puhul vali **kõik** sellega lingitud punktid (st üks või kaks).
- Less (vähem): kui iga valitud kontrollpunkti puhul on valitud ka **kõik** sellega lingitud punktid, jäta see punkt valituks. Kõigi teiste valitud kontrollpunktide valik tühista.

See tähendab kahte asja:

- Esiteks, kui valitud on **kõik** kõvera kontrollpunktid, ei juhtu midagi (sest vähem (Less) puhul on valitud kõik lingitud punktid ja rohkem (More) ei saa enam loomulikult rohkem punkte lisada). Samamoodi juhtub siis, kui valitud ei ole ühtegi kontrollpunkti.
- Teiseks ei liigu need tööriistad kunagi kõverast "väljapoole" (nad ei "hüppa" kunagi teise samas objektis asuvasse kõverasse).

Kõveraga deformeerimine (*Curve Deform*)

Kõveraga deformeerimise (Curve Deform) abil saab kerge vaevaga võret deformeerida. Võreobjekti allutamisel kõvera tüüpi objektile saab võret deformeerida seda mööda kõvera peatelge liigutades. Nii saab sooritada keerukaid rada järgivaid animatsioone nagu näiteks paberi liikumine printeris, filmi liikumine kaameras, vee liikumine kanalis.

Kõveraga deformeerimine (Curve Deform) toimib globaalsetel XYZ-telgedel. See tähendab, et võret liigutamisel kõvera peatelje suunas muutub vastavalt kõvera kujule ka võre deformatsioon. Võre liigutamisel kõvera peateljega risti liigub see kõverale lähemale või sellest kaugemale. Vaikimisi on Blenderis peateljeks määratud Y-telg. Objekti nihutamisel kõvera otstest edasi toimub objekti deformeerimine vastavalt kõvera viimase segmendi suunale.

Kui kõver on 3D-kõver, kasutatakse deformeerimisel ka kõvera kontrollpunktide kaldenurka (Tilt). Kahanda/laienda (Shrink/Fatten) omadust muutes saab muuta deformeeritava objekti paksust vastava kontrollpunkti piirkonnas.



Soovitus

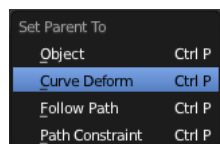
Enne kõveraga deformeerimise asumist on soovitatav kõver ja võre paigutada samasse asukohta. Nii on deformeerimist kõige kergem teha.

Kasuta töötlejaid (*modifiers*)!

Kõveraga deformeerimise (Curve Deform) sõltuvus on nüüd saadaval ka töötlejana ja selle nimi on [kõveratöötleja \(Curve\)](#).

Kõveratöötleja (Curve) toimib täpselt samal viisil, erinevus on vaid selles, et töötlejas saab peatelje ise määrata ning nupud Track X/Y/Z asja enam ei mõjuta. On olemas selline tore asi nagu võimalus võreobjekti kõverdada, nii et kõverdamine mõjutaks ainult määratud tipugruppe.

Kasutajaliides

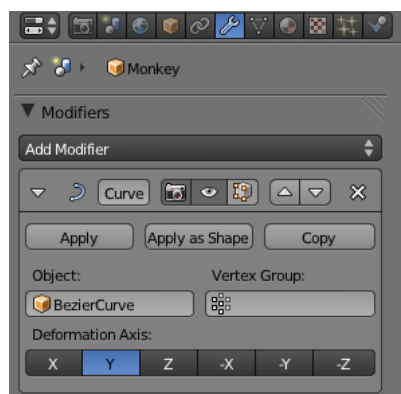


Allutamise menüü

Make Parent.

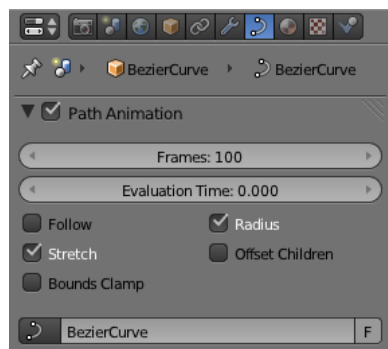
Kui sa vajutad objekti kõverale allutamiseks klahvikombinatsiooni CtrlP, avaneb allutamise menüü Make Parent.

Valides allutamise tüübiks Curve Deform, lisatakse võrele kõvera abil deformeerimise töötleja.



Animatsiooni sätete paneel Anim settings.

Domineeriva telje seadistused on võreobjektil paika pandud. Vaikimisi on Blenderis peateljeks Y-telg. Seda saab muuta, vajutades Track X, Y või Z nuppu paneelil Anim (vt pilti *Animatsiooni sätete paneel Anim settings*) objekti (Object) kontekstis (F7).



Kõverate ja pindade paneel Curve and Surface.

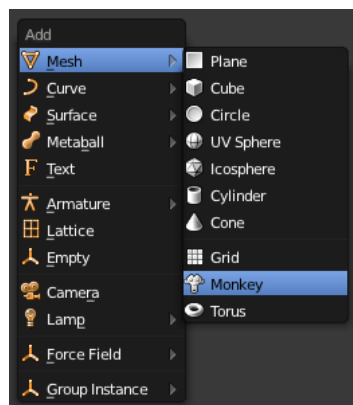
Tsükliline ehk suletud kõver toimib deformeerimisel tsüklilise deformeerijana. Kui ühe kõvera tüüpi objekti sees on mitu kõverat, jälgib sellele allutatud objekt vaid esimest kõverat.

Venitamise (Stretch) valik võimaldab võreobjekti venitamist ja kokkusurumist kogu kõvera ulatuses. See valik on saadaval Object Data saki Path Animation sektsiooni all siis, kui valitud on deformeeriv kõver. Vaata pilti *Kõverate ja pindade paneel Curve and*

Surface.

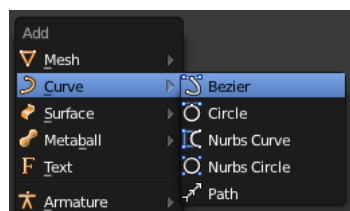
Näidis

Teeme lihtsa näite:



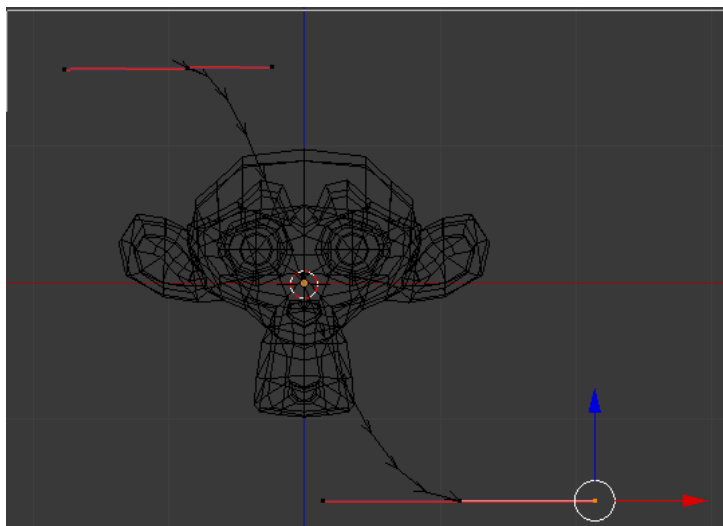
Lisa pärdik!

- Kustuta vaikekuubik ja lisa pärdik (⇧ ShiftA » Add » Mesh » Monkey, vt pilti *Lisa pärdik!*)!
- Muutmisrežiimist väljumiseks vajuta ⇧ Tab-klahvi.



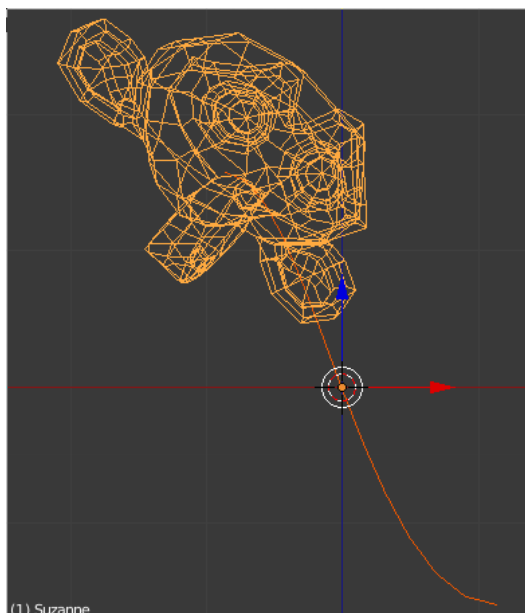
Lisa kõver.

- Nüüd lisa kõver (⇧ ShiftA » Add » Curve » Bezier Curve, vt pilti *Lisa kõver*).



Kõvera muutmine.

- Sisene muutmisrežiimi ja muuda kõver selliseks, nagu on pildil "Kõvera muutmine"; seejärel välju muutmisrežiimist.



(1) Suzanne
Pärdik kõveral.

- Nüüd võid sa pärdiku deformeerimiseks kasutada uut kõveraga deformeerimise töötlejat:
 - Vali pärdik (RMB).
 - Lisa sellele töötlejate paneelist Modifiers kõveratöötleja (Curve).
 - Objektiväljale trüki sisse kõvera nimi (vaikimisi on see "Curve"). Sealt võid muuta ka deformatsiooni peatelge.
- Võid kasutada ka vana ja iganenud meetodit (pane tähele, et see tekitab "virtuaalse" töötleja):
 - Vali pärdik (RMB) ja siis lisa valikusse kõver (Shift RMB .
 - Allutamise menüü Make Parent avamiseks vajuta klahvikombinatsiooni CtrlP.
 - Vali Curve Deform (vt pilt *Allutamise menüü Make Parent*).
- Pärdik peaks olema kõvera suhtes asetatud nii, nagu pildil *Pärdik kõveral*.
- Nüüd, märkides RMB abil pärdiku ja seda Y-suunal (vaikimisi domineeriv telg) liigutades (G) deformeerub pärdik kenasti piki kõverat.



Nipp

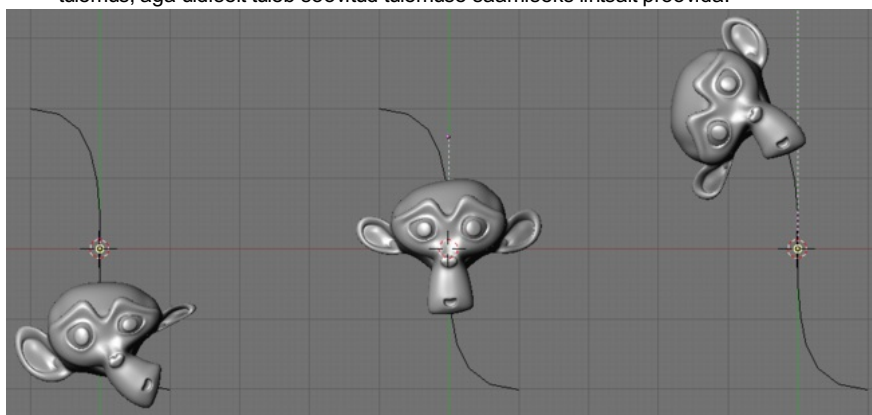
Liigutamisel (pärast G klahvi vajutamist) saad liikumise piirata vaid ühe teljega, vajutades klaviatuuril vastava telje tähisega klahvi. Ehk siis X-, Y- või Z-klahvi vajutades.

- Pildil (*Pärdiku deformatsioonid*) on näha pärdiku erinevad deformatsioonid sõltuvalt asukohast kõvera suhtes. Ilusama pildi saamiseks on objektile lisatud tükeldamise (SubSurf) töötleja tasemega **2** ja lisaks sellele on objektile rakendatud silumine (Set Smooth).



Nipp

Pärdiku liigutamisel mõnes muus suunas kui peatelg, on tulemuseks imelikud deformatsioonid. Mõnikord see ongi soovitud tulemus, aga üldiselt tuleb soovitud tulemuse saamiseks lihtsalt proovida!



Pärdiku deformatsioonid.

Kõvera eendamine

See osa räägib kõvera eendamise meetoditest ehk kõveratele paksuse andmisest ning sellest, kuidas seda paksust raja pikkuse ulatuses kontrollida.

Eendamine (*Extrusion*)

Mode: Objektirežiim Object või muutmisrežiim Edit

Panel: Kõver ja pind Curve and Surface (muutmise kontekst Editing, F9)

Ah! Eendamine! Tõenäoliselt kõige huvitavam kõveratega modelleerimise tööriist, eriti siis, kui kasutate kantimise/koondamise/kallutamise/raadiuse sätteid. Pane tähele, et sellel pole midagi pistmist eendamise käsuga Extrude (E), mida kirjeldati [eelmisel leheküljel](#)!

Vaatame erinevaid sätteid vastavalt nende poolt läbiviidava tegevuse ulatusele:

Width (laius)

See määrab kõvera eendatud "serva" asukoha kõvera suhtes. Suletud kahemõõtmeliste kõverate puhul (loe altpoolt) on seda suhteliselt lihtne mõista – kui laius Width on suurem kui **1.0**, on eendatud maht laiem, laiuse väärtuse **1.0** puhul järgib serv kõverat väga täpselt ning laiuse väärtuse puhul, mis on väiksem kui **1.0**, on maht kitsam. Sama põhimõtte töötab ka avatud kahemõõtmeliste ja kolmemõõtmeliste kõverate puhul, kuid see, kuidas määratakse kõvera "välimine" ja "sisemine" pool, näib olevat natuke imelik.

Selle mõju eendatud "kandidud" objektidele on samasugune.

Tilt (kalle)

See säte – kahjuks ei näe sa Blenderis kusagil selle väärtust – määrab iga kõvera punkti puhul "väändenurga", seega on see oluline ainult kolmemõõtmeliste kõverate puhul!

Seda saab määrata kalde teisendamise tööriista abil (Tilt, T või menüüst Curve » Transform » Tilt) ning samuti saad sellele tagasi anda vaikeväärtuse (st risti algse kõvera tasapinnaga), kasutades valikut AltT (või menüüd Curve » Control Points » Clear Tilt).

NURBSide puhul interpoleeritakse kallet alati sujuvalt. Kuid Bezier' puhul võid kõvera tööriistade paneeli Curve Tools kalde interpolatsiooni ripploendist Tilt Interpolation valida interpoleerimise algoritmi (leidad sealt klassikalised võimalused lineaarne (Linear), kardinaalne (Cardinal), Bezier (B Spline) ja sujuv (Ease)).

Lihtne eendamine

Vaatame kõigepealt kõverate "lihtsat" eendamist ning jätame kantimise/koondamise lisaobjektid kõrvale.

Extrude (eenda)

See määrab eendamise laiuse (või kõrguse). Reaalne suurus oleneb loomulikult aluseks oleva objekti mõõtkavast, kuid mõõtkava puhul väärtusega 1 eendab Extrusion väärtus **1.0** kõverat mõlemas suunas ühe Blenderi ühiku võrra kõvera tasapinnaga risti asetseval teljel (kolmemõõtmeliste kõverate eritingimuste kohta loe altpoolt).

Kui väärtus on **0.0**, siis "lihtsat" eendamist ei toimu!

Bevel Depth (kandi sügavus)

See lisab eendamisele kandi. Loe selle mõju kohta altpoolt.

Pane tähele, et kant muudab eendamise tulemuse laiemaks ja kõrgemaks.

Kui väärtus on **0.0**, siis kantimist ei toimu (maksimaalne väärtus: **2.0**).

Bev Resol (kandi resolutsioon)

Määrab resolutsiooni kantidel, mis on loodud kandi sügavuse (Bevel Depth) väärtusega, mis on nullist kõrgem.

Kui väärtus on **0** (vaikeväärtus), on kant lihtne "lame" pind.

Kõrgemad väärtused siluvad ja kumerdavad kanti ning on seetõttu samased kõvera enda resolutsiooni sätetega.

Meil on ka kolm võimalike tulemuste alamklassi, mis sõltuvad sellest, kas kõver on avatud, suletud või kolmemõõtmeline:

Avatud kahemõõtmeline kõver

Eendamised loovad kõvera kuju järgiva "seina" või "riba". Kui kasutate kandi sügavust (Bevel Depth), muutub sein liumäe või rentslisarnaseks. Pea meeles, et kandi suund on mõnikord imelik ja etteaimamatu ning sageli vastupidine sellele, mille soovite saavutada samasuguse suletud kõvera puhul. Seda suunda saad tagurpidi keerata, [muutes kõvera suunda](#).

Niimoodi võid näiteks kiirelt simuleerida mööda keerulist liumäge alla veerevat kivikuuli, kombineerides eendatud ja kandidud kõvera ning kera, millele on lisatud kõveraga seotud rada järgiv (Follow Path) piiraja.

Suletud kahemõõtmeline kõver

See on eeldatavasti kõige kasulikum olukord, sest sellega saad kiirelt luua mahu, kus (vaikimisi) kaks lamedat ja paralleelset pinda täidavad kaks eendatud "seina" külge. Võid neist külgedest ühe või mõlemad eemaldada, lülitades 3D nupu kõrval olevad tagakülje (Back) ja/või esikülje (Front) lülitid välja.

Kant on antud juhul alati õigesti orienteeritud ning seeläbi saad mahu "servad" siledaks ajada.

Kolmemõõtmeline kõver

Selliste kõverate puhul ei ole sellel, kas kõver on suletud või mitte, üldse tähtsust – eendatud kolmemõõtmelisest kõverast ei teki kunagi mahtu, vaid ainult sein või riba nagu avatud kahemõõtmeliste kõverate puhulgi.

Kuid kolmemõõtmeliste kõverate puhul on üks lisavõimalus: kontrollpunktide kalle (Tilt, loe ülaltpoolt). See paneb riba kõvera ulatuses väänlema, luues näiteks Möbiuse lehe!

Keerulisem eendamine

Need eendid kasutavad ühte või kahte lisa kõverat, et luua väga keerulisi orgaanilisi kujundeid.

Seda tüüpi eendamise sisselülitamiseks pead kirjutama selle kõvera, mida kavatsed kasutada oma eendamise "selgroona", kantobjekti väljale BevOb sobiva kõveraobjekti nime. "Kantimise" kõver määrab eendatud objekti ristlõike. Pole oluline, kas väljas BevOb olev kõver on kahe- või kolmemõõtmeline, kuid kui see on suletud, luuakse "torujas" eendamine ning muul juhul on tulemuseks liumäge või rentslit meenutav objekt.

Objekt eendatakse kõigi selle seesmiste kõverate täies pikkuses. Vaikimisi on eendamise laius konstantne, kuid sul on selle muutmiseks kaks võimalust: kontrollpunktide raadiuse väärtus Radius ning "koondamise" objekt.

Punktide raadiust (Radius) saab muuta raadiuse kahandamise/paksendamise teisendustööriistaga Shrink/Fatten Radius (AltS või menüüst Curve » Transform » Shrink/Fatten Radius) või eriomaduste menüü Specials (W) raadiuse määramise valikuga Set Radius.

Ka selle puhul ei saa sa valitud kontrollpunkti raadiust kahjuks kusagilt näha.

Raadiusega saad sa otseselt kontrollida eendamise laiust "selgroo" kõvera ulatuses. Nagu kalde puhulgi (loe ülaltpoolt), saad kõvera tööriistade paneeli Curve Tools raadiuse interpoleerimise ripploendist Radius Interpolation valida Bezier' kõverate puhul kasutatava interpolatsiooni algoritmi.

Kuid sul on ka teine ja täpsem võimalus: "koondamise" objekt (*taper*). Nagu ka "kantimise" objekti puhul, määrad sa selle nime peamise kõvera koondamise objekti väljas TaperOb – see peab olema *avatud kõver*. Koondamise kõverat arvutatakse piki *kohalikku X-telge* ning *kohalik Y-telg* määrab selle laiuse. Lisaks pane tähele:

- Koondamine määratakse eendatud objekti kõigile kõveratele eraldi.
- Kasutatakse ainult koondamisobjekti TaperOb esimest kõverat ning seda ka siis, kui objekt koosneb mitmest eraldi segmendist.
- Mõõtkava algab esimesest vasakpoolsest kontrollpunktist ning liigub mööda kõverat viimase kontrollpunktini paremal.
- Negatiivne mõõtkava (koondamiskõvera kohalik negatiivne Y) on samuti võimalik. Kuid sellega võivad renderdamisel artefaktid tekkida.
- See muudab tavaliste eendamiste mõõtkava laiust vastavalt koondamiskõverale, mis tähendab, et koondamiskõvera teravaid nurki pole hästi näha. Pead selleks kõvasti tõstma aluskõvera resolutsiooni (DefResolU).
- Suletud kõverate puhul toimib koondamisobjekt TaperOb terve kõvera piirjoone ulatuses ja mitte ainult objekti pikkuse suhtes, seega muutub ka eendamise sügavus. Sellistel puhkudel on kasulik hoida koondamiskõvera TaperOb mõlema otsa suhteline kõrgus samana, et kõvera ühenduspunkt (koht, kus kõvera lõpp-punkt ühendub algusega) oleks sujuva üleminekuga.

Ning viimaks - kolmemõõtmeliste "selgroo" kõverate puhul määrab kontrollpunktide kalle (Tilt) eendatud "kandi" väändumise mööda kõverat!

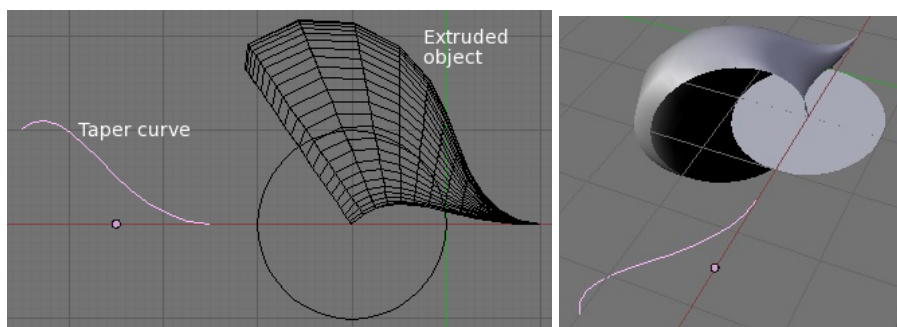
Näited

Tuleb teha: lisa mõned "lihtsa" eendamise näited.

Tuleb teha: lisa mõned näited "kandiga" eendamisest, mis kasutavad raadiust (Radius).

Koondame lihtsat ringikujulist kõverat, kasutades selleks koondamiskõverat. Lisa kõver, välju muutmisrežiimist, lisa teine kõver (suletud nagu ring), pane selle nimeks "KantKover" ning sisesta see nimi esimese kõvera kantobjekti välja BevOb (muutmise kontekst Editing F9, kõvera ja pinna paneel Curve and Surface). Meil on nüüd toru. Lisa objektirežiimis kolmas kõver ning nimeta see "KoondKover". Muuda vasakpoolset kontrollpunkti, tõstes seda 5 ühiku võrra.

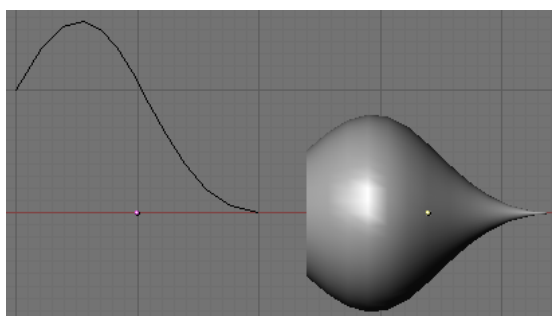
Mine nüüd tagasi muutmise [konteksti](#) Editing ning muuda esimese kõvera koondobjekti välja TaperOb ([kõvera ja pinna paneelis Curve and Surface](#)), et selle sisuks oleks "KoondKover" nime saanud kõver. Kui sa nüüd selle kinnitad, siis lisatakse koondamine koheselt ning selle tulemus on näha pildil (*Koondamisega eendatud kõver*).



Koondamisega eendatud kõver.

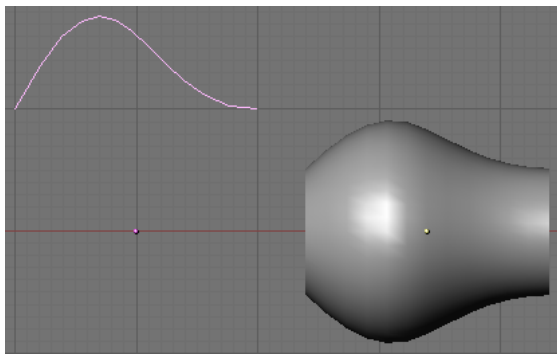
Koondamine tahkes režiimis.

Näed, et **eendamisobjektile** on lisatud **koondamise kõver**. Pane tähele, kuidas toru maht kahaneb vastavalt sellele, kuidas koondamise kõver liigub vasakult paremale. Kui koondamise kõver liiguks kohalikust Y-teljest allapoole, muutuks torus sisemus välimuseks (lõikuks iseendaga) ning selle tulemuseks oleksid renderdamisel artefaktid. Loomulikult võib see olla see tulemus, mida sina kui kunstnik soovid!



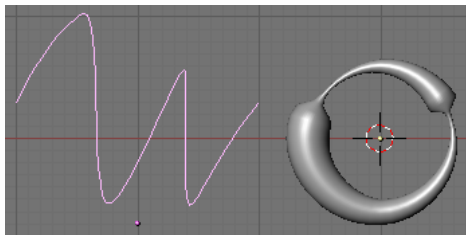
Koondamise näide 1.

Pildil (*Koondamise näide 1*) näed sa selgelt, millist mõju avaldab koondamiskõver parempoolsele kõverobjektile. Antud juhul on vasakpoolne koondamiskõver lähemal objekti keskpunktile ning tulemuseks on väiksem paremal olev kõverobjekt.



Koondamise näide 2.

Pildil (*Koondamise näide 2*) on vasakpoolses koondamiskõveras liigutatud üks kontrollpunktidest keskpunktist eemale ning tulemuseks on laiem paremal asuv kõverobjekt.

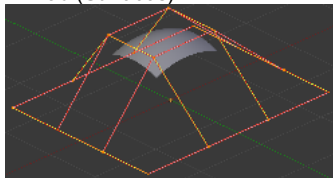


Koondamise näide 3.

Pildil (*Koondamise näide 3*) näeme me seda, kuidas ebaregulaarse kujuga koondamiskõver mõjutab ringikujulist kõverat.

Tuleb teha: lisa mõned näited "kandiga"eendamisest, mis kasutavad kallet (Tilt).

Pinnaid (Surfaces)



Pind.

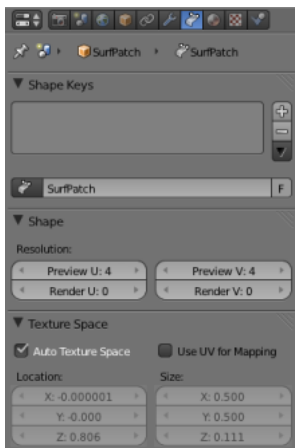
Kõverad on kahemõõtmelised objektid ning pinnad on nende laiendused kolmemõõtmelisse ruumi. Pane tähele, et Blenderis on kasutada ainult NURBS-pinnad, Bezier' omi pole (küll aga on olemas Bezier' tüüpi sõlmed, loe nende kohta altpoolt), samuti pole polügonaalset (nende asemel on sul olemas võred!) pindu. Kuigi kõverad ja pinnad jagavad sama objekti tüüpi (samamoodi tekstiobjektid), pole nad samad asjad, näiteks ei saa samas objektis olla nii kõveraid kui ka pindu.

Kuna pinnad on kahemõõtmelised, on neil kaks interpolaatsiooni telge: U (nagu kõveratel) ja V. Tähtis on mõista, et sa saad määrata neile kahele dimensioonile interpoleerimise reeglid (sõlm, järk, resolutsioon) eraldi (kõigi sätete puhul U ja V telgedele eraldi).

Sa võid küsida: "Aga need pinnad näivad kolmemõõtmelised, miks on nad ainult kahemõõtmelised?" Et objekt oleks kolmemõõtmeline, peab sel olema "maht", ning pinnal, isegi suletud pinnal, ei ole mahtu - see on lõpmatuseni õhuke. Kui sel oleks maht, oleks pinnal paksus (selle kolmas mõõde). Seega on see ainult kahemõõtmeline objekt ning sel on ainult kaks interpolaatsiooni mõõdet ehk telge ehk koordinaati (kui sa tunned natuke matemaatikat, kujuta endale ette mitte-eukleidilist geometriat – pinnad ongi kahemõõtmelised mitte-eukleidilised tasapinnad). Rohkem "reaalsest elust" võetud näide oleks see, et sa võid paberilehe silindri loomiseks kokku keerata, kuid isegi kui see "tekkitab" mahu, jääb leht ise (peaaegu) kahemõõtmeliseks objektiks!

Tegelikult on pinnad väga sarnased sellele tulemusle, mida saad siis, kui [eendad kõverat](#) (mina isiklikult arvan, et peaks olema võimalik konverteerida eendatud kõverat pinnaks, vähemalt NURBS-kõverast tehtud eendamist - kuid Blender seda praegu ei suuda).

Pinna tööriistade leidmine



Pinna tööriistad.

Muutmise kontekstis Editing olevad paneelid on samad, mis [kõverate puhul](#), kuid vähemate valikutega. Ning nagu tavaliselt, on sul 3D-vaate päistes valimise Select ja pinna Surface menüüd ning lisavalikutega hüpikmenüü Specials (W).

Kuvamine

Pindade ja NURBS-kõverate puhul pole kuvamises erilist vahet, välja arvatud see, et u suunda näitavad kollased ruudustiku jooned ja v suunda roosad ruudustiku jooned, nagu näed ka pildil (*Pind*).

Kontrollpunkte saad [peita ja näidata](#) nagu kõverate puhulgi ning sul on samad kuvamise režiimid nagu kõvera tööriistade paneelis [Curve Tools](#).

Pinna struktuur

Paljud [kõverate](#), eriti [NURBS-kõverate](#) mõisted kehtivad otseselt ka NURBS-pindade puhul, näiteks kontrollpunktid, järk (Order), kaal (Weight), resolutsioon (Resolution) jne. Siin räägime ainult erinevustest.

Väga oluline on mõista NURBS-kõverate ja NURBS-pindade vahet - esimestel on üks mõõde, teistel kaks. Blender kohtleb NURBS-pindu ja NURBS-kõveraid sisemiselt täiesti erinevalt. On mitu omadust, mis neid eristavad, kuid kõige olulisem on see, et NURBS-kõveratel on ainult üks interpolaatsiooni telg (U), aga NURBS-pindadel on kaks interpolaatsiooni telge (U ja V).

Samas võid sa luua kõveratest "kahemõõtmelisi" pindasid (kasutades [eendamise tööriistu](#) või vähemal määral ka kahemõõtmeliste suletud kõverate täitmist rakendades). Ja sa saad pindadest luua "ühemõõtmelisi" kõveraid, sest NURBS-pind, millel on ainult üks rida (kas U või V suunas) kontrollpunkte, tekitab kõvera.

Visuaalselt saad sa neid eristada, sisenedes muutmisrežiimi Edit ja vaadates 3D-akna päist: seal on ühe menüü valikuna kas "pind (Surface)" või "kõver (Curve)". Samuti saad sa pinna loomiseks terve NURBS-pinna kõvera [eendada](#), kuid tavalise NURBS-kõveraga sa seda teha ei saa (me räägime "tavalisest" eendamise tööriistast Extrude, mis aktiveerub kiirvalikuga E, mitte suhteliselt spetsiifilisest kõvera eendamise tööriistast – jah, ma tean, et see tekitab segadust).

Kontrollpunktid, read ja ruudustik

NURBS-pindade kontrollpunktid on samad, mis NURBS-kõveratel. Kuid nende asetus on suhteliselt piirav. "Segmendi" mõiste kaob

ning seda asendavad "rida" ja üldine "ruudustik".

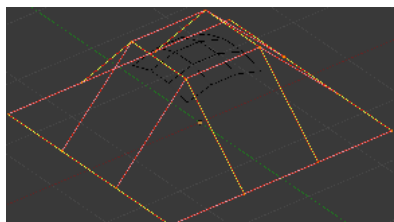
"Rida" on kontrollpunktide hulk, mis loob ühes interpolatsiooni suunas "joone" (see on natuke sarnane võrede mõistega [silmuslõiked](#) (ehk [servatsükli](#))). Seega on NURBS-pinnal "U-read" ja "V-read". Peamine on see, et *kõigil ühte tüüpi ridadel (U või V) on sama arv kontrollpunkte*. Iga kontrollpunkt kuulub ainult ühte U-ritta ja ühte V-ritta.

Kõik see kokku moodustab "ruudustiku" ehk "puuri", mille kuju määrab NURBS-pinna kuju. See sarnaneb natuke [sõrestikule](#).

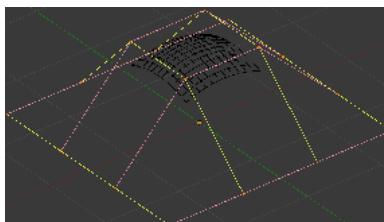
Selle mõistmine on väga oluline: sa ei saa NURBS-pinnale lisada ühte kontrollpunkti, vaid pead lisama korraga terve U- või V-rea (praktikas kasutad sa nende lisamiseks tavaliselt eendamise tööriista või duplitseerimist), mis sisaldab täpselt samapalju punkte kui teised. Samuti tähendab see, et saad kaks erinevat pinnatükki "ühendada" vaid siis, kui vähemalt üks nende ridadest sobib teise tüki ridadega.

Pinna resolutsioon

Nagu [NURBS-kõverate](#) puhulgi, määrab resolutsioon Resolution pinna detailsuse. Mida kõrgem on resolutsioon, seda detailsem ja siledam pind on. Mida madalam on resolutsioon, seda konarlikum on pind. Sa saad kasutada kahte resolutsiooni sätet, mis vastavad kahele interpolatsiooni teljele (U ja V). Pane tähele, et erinevalt kõverate resolutsioonist on sul siin ainult üks resolutsioon (kõvera tööriistade Curve Tools paneelis asuvad Resol U ja V väljad).



Resolutsioon 1×1.



Resolutsioon 3×3.

Pildil (*Resolutsioon 1×1*) on näide, kus nii u kui ka v resolutsioon on 3. Pildil (*Resolutsioon 3×3*) on näide, kus nii u kui v resolutsioon on 12.



Resolutsiooni paneel.

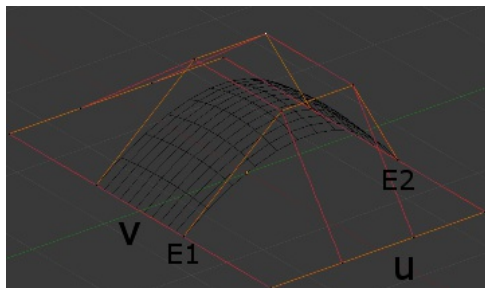
Saad määrata eelvaate ja renderdamise resolutsiooni eraldi, et vaateaken ei muutuks aeglaseks, kuid renderdamise tulemus oleks siiski hea.

Suletud ja avatud pinnad

Nagu kõveradki, saavad pinnad olla mõlemas suunas eraldi kas suletud (tsükilised) või avatud, mistõttu saad lihtsalt luua toru, pontsiku või kera kujusid ning neid kuvatakse muutmisrežiimis Edit "tahketena". See muudab pindadega töötamise suhteliselt lihtsaks.

Sõlmed

Nagu [NURBS-kõverate](#) puhulgi, on NURBS-pindadel kaks sõlmevektorit, üks U- ja teine V-teljele. Ning nad võivad taas kord olla kas ühtlased (`{Literal|Uniform}`), lõpp-punkti (Endpoint) või Bezier' (Bezier) tüüpi ja nende omadused vastavad kõverate omadele. Ja nagu kõverate puhul, mõjutavad need sätted (vastavas suunas) vaid avatud pindu.



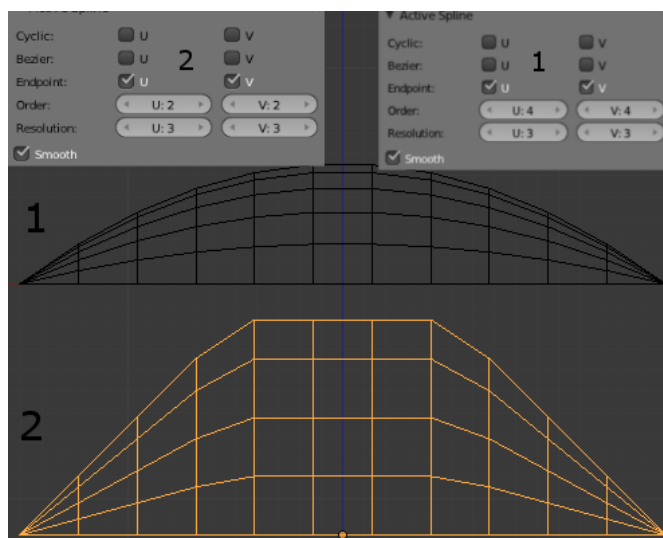
Lõpp-punkt U

Pildil (*Lõpp-punkt U*) on U interpolatsiooni telg tähistatud kui "u" ja V interpolatsiooni telg tähistatud kui "v". u interpolatsiooni telje tüübiks on määratud lõpp-punkt (Endpoint) ning seetõttu ulatub pind mööda u interpolatsiooni telge väliste külgedeni "E1" ja "E2".

Et pind ulatuks kõigi väliste külgedeni, tuleks ka v telje tüüp määrata lõpp-punktiks (Endpoint).

Järk

Ka seekord on see omadus sama nagu [NURBS-kõverate](#) puhul ning see määrab, kui palju kontrollpunkte pinna kuju kõveruse arvutamisel arvesse võetakse. Kõrgete järkude Order puhul, pildi osa (1), tõmbub pind kontrollpunktidest eemale ja tulemus on siledam – juhul kui [resolutsioon](#) on piisavalt kõrge. Madalate järkude Order puhul, pildi osa (2), järgib pind kontrollpunkte ning tulemuseks on pind, mis kipub jälgima ruudustiku puuri.

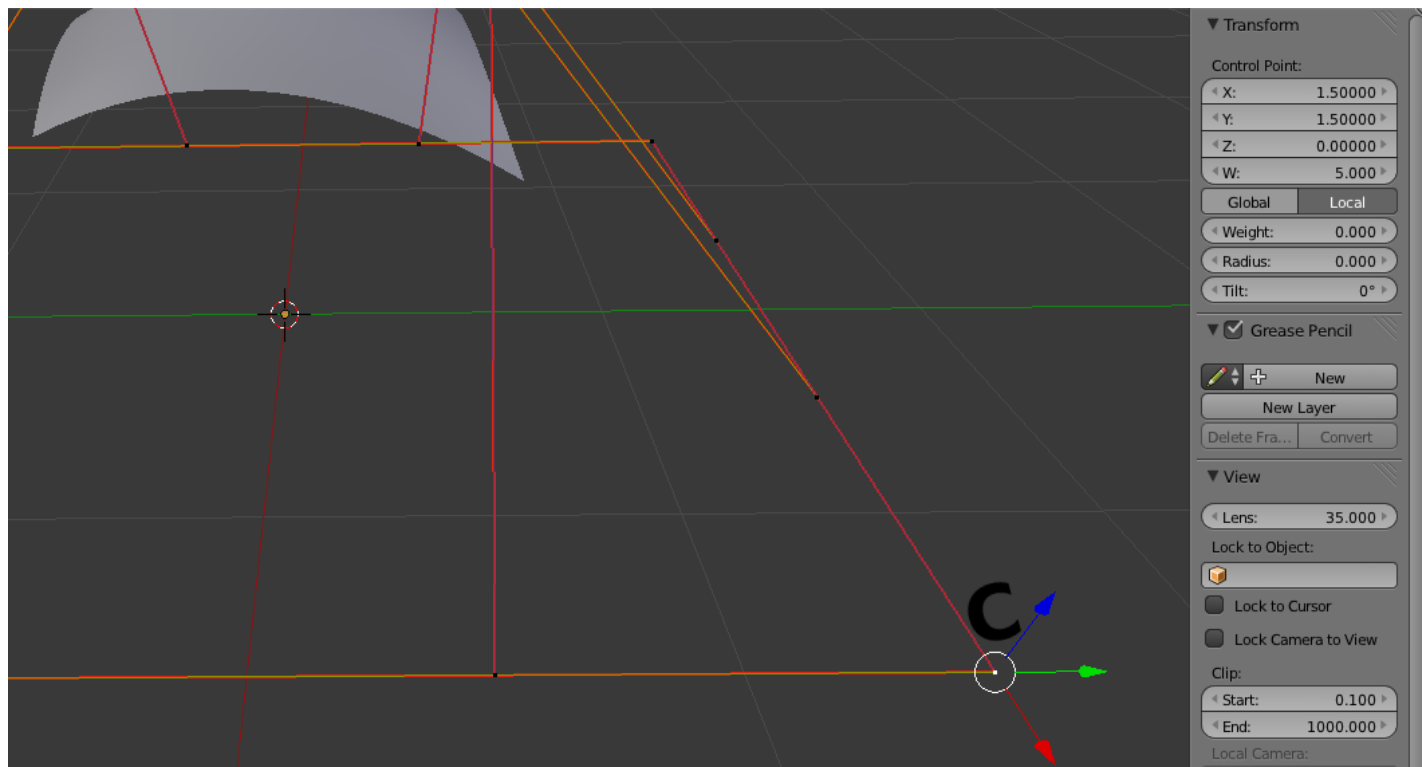


Teist ja neljandat järku pinnad.

Illustreerimise eesmärgil on nii pildil (*Neljandat järku pind*) kui ka pildil (*Teist järku pind*) määratud sõlmevektorite tüübiks lõpp-punkt (Endpoint), mistõttu ulatub pind kõigi väliste külgedeni.

Mõlemale interpolatsiooniteljele saad sa järjestuse määrata eraldi ning nagu ka kõverate puhul ei saa see olla madalam kui **2** ega kõrgem kui **6** või vastaval teljel olev kontrollpunktide arv.

Kaal



Pinna kaal 5.

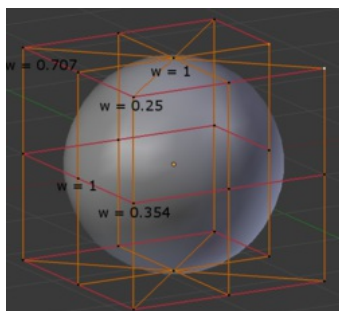
No mis sa arvad? Jah, see töötab täpselt samamoodi nagu [NURBS-kõverate](#) puhul! Kaal (Weight määrab, palju iga kontrollpunkt kõverat "tõmbab").

Pildil (*Pinna kaal 5*) on tähisega "c" kontrollpunkti kaaluks Weight määratud **5.0**, samas kui kõigi teiste oma on vaikimisi **1.0**. Nagu sa näha võid, *tõmbab* too punkt pinda enda poole.

Kui kõigil kontrollpunktidel on sama kaal Weight, siis tühistavad nad põhimõtteliselt üksteise väärtusi. Kaalude erinevus on see, mis sunnib pinda kontrollpunktide poole või neist eemale liikuma.

Iga kontrollpunkti kaal on nähtav teisenduste omaduste paneelis [Transform Properties](#) (N) olevas väljas W (mitte väljas Weight).

Eelseadistatud kaalud



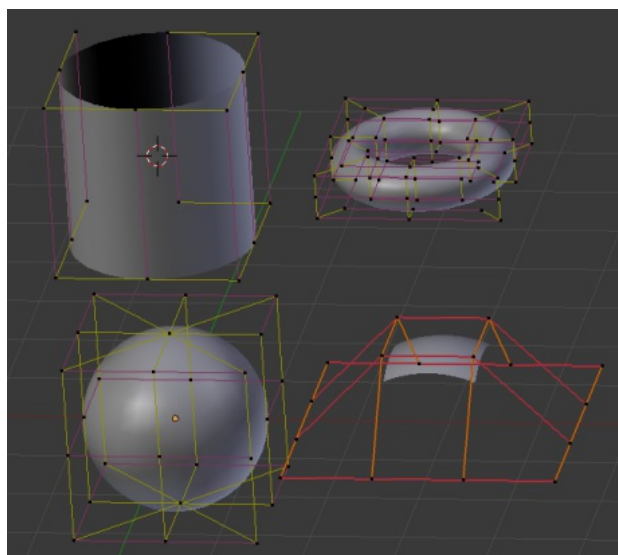
Kerakujuline pind.

NURBSid saavad luua puhtaid vorme nagu ringid, silindrid ja kerad (pea meeles, et Bezier' ring ei ole puhas ring). Puhaste ringide, kerade või silindrite loomiseks pead määrama kontrollpunktidele kindlad kaalud, milledest mõned on eelseadistustena kõvera tööriistade paneelis (Curve Tools) ära toodud (alumine parempoolne nurk).. See ei ole kohe selgelt mõistetav ning sa peaksid enne nende proovimist lugema NURBSide kohta täpsemalt.

Me nägime ühemõõtmeliste [NURBS-kõverate](#) juures, kuidas saab luua ringi. Vaatame nüüd, kuidas saab kahemõõtmeliste pindadega luua kera. Põhimõte on sama – pane tähele, et kera loomiseks vajaminevad neli erinevat kaalu (**1.0**, **0.707** = $\text{ruutjuur}(0.5)$, **0.354** = $\text{ruutjuur}(2)/4$ ja **0.25**) on kõvera tööriistade paneelis [Curve Tools](#) eelseadistustena olemas.

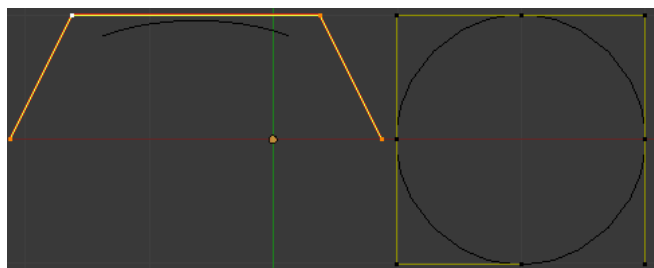
Primitiivid

Et sul pindade loomise puhul algust lihtsamaks teha, on pinna lisamise menüüs Add » Surface olemas neli NURBS-pinna eelseadistust: NURBS-pind (NURBS Surface), NURBS-toru (NURBS Tube), NURBS-kera (NURBS Sphere) ja NURBS-toroid (NURBS Torus).



NURBS-pinna primitiivid.

On olemas ka kaks NURBS-pinna kõvera eelseadistust (mille puhul on igal V-real ainult üks kontrollpunkt): NURBS-kõver (NURBS Curve) ja NURBS-ring (NURBS Circle).



NURBS-kõvera primitiivid.

Pane tähele, et ringikujulise NURBS-pinna keskel olevat auku ei täideta kunagi, erinevalt selle NURBS-kõvera vastest.

Pinna valimine

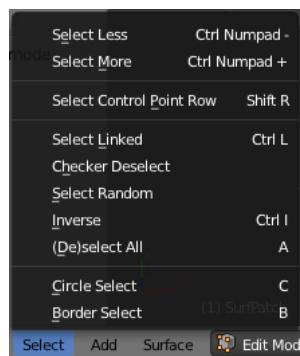
Pinna valimine muutmisrežimis Edit on väga sarnane [NURBS-kõvera valimisele](#). Lihtsad tööriistad on samad mis [võrede](#) puhul, seega saad valida kontrollpunkti lihtsalt klahviga LMB , lisada valikusse klahviga ⇧ Shift LMB , valida piirdkastiga B ja nii edasi.

L (või CtrlL) lisab valikusse hiirekursorile lähima kontrollpunkti ning kõik sellega lingitud punktid, st kõik samale pinnale kuuluvad punktid.

Valikumenüü (*Select*)

Valikumenüü Select (3D-vaate päises) on veelgi lihtsam kui kõverate oma...

Kõigil valikutel on sama tähendus ja toimimisviis kui [objektirežimis](#) (ning muutmisrežimis Edit piirdkastiga valimisest (Border Select) on räägitud juba [siin](#)).



Iga n. (*Every Nth*)

Mode: Muutmisrežiim Edit

Hotkey: Puudub

Menu: Select » Every Nth

See on sama võimalus nagu [[Doc:ET/2.6/Manual/Modeling/Curves/Editing#Iga n. (*Every Nth*)|kõverate valimise puhul]]. Kuigi parameetri N ("valiku samm") käitumisest on kahemõõtmelisena NURBS-pinna "puuris" suhteliselt raske aru saada...

Kontrollpunktide rida (*Control Point Row*)

Mode: Muutmisrežiim Edit

Hotkey: ⇧ ShiftR

Menu: Select » Control Point Row

See võimalus töötab natuke võrede [serva silmuse valiku](#) samaselt, sest valib terve [rea](#) kontrollpunkte sõltuvalt aktiivsest (viimasena valitud) kontrollpunktist. Esimene kord, kui sa kasutad kombinatsiooni ⇧ ShiftR, *lisatakse hetkevalikule* V-rida, mis läbib (sisaldab) aktiivset punkti. Kui sa kasutad seda kiirkombinatsiooni uuesti, siis lülitub valik selle punkti U- ja V-ridade vahel, *eemaldades valikust kõik muu*.

Rohkem ja vähem (*More/Less*)

Mode: Muutmisrežiim Edit

Hotkey: Ctrl+ NumPad/Ctrl- NumPad

Menu: Select » More/Less

Need kaks valikut on võrdväärsed ja väga sarnased [võre valiku omadele](#). Nende eesmärk, sõltuvalt parajasti valitud kontrollpunktidest, on valikut suurendada või vähendada.

Algoritm on sama mis võrede puhul:

- More (rohkem): iga kontrollpunkti puhul vali **kõik** sellega lingitud punktid (st kaks, kolm või neli).
- Less (vähem): kui iga valitud kontrollpunkti puhul on valitud ka **kõik** sellega lingitud punktid, jäta see punkt valituks. Kõigi teiste valitud kontrollpunktide valik tühistab.

See tähendab kahte asja:

- Esiteks, kui valitud on **kõik** pinna kontrollpunktid, ei juhtu midagi (sest vähem (Less) puhul on valitud kõik lingitud punktid ja rohkem (More) ei saa enam loomulikult rohkem punkte lisada). Samamoodi juhtub siis, kui valitud ei ole ühtegi kontrollpunkti.
- Teiseks ei liigu need tööriistad kunagi pinnast "väljapoole" (nad ei "hüppa" kunagi teise samas objektis asuvasse pinda).

Pinna muutmise

Pinna muutmisel on veelgi vähem tööriistu ja valikuid kui kõvera puhul ning nende vahel on palju sarnasusi... Seega räägib see leheküljel (või proovib rääkida) kõigist teemadest, alates pindade muutmise algteadmistest, kuni keerukamate teemadeni nagu topoloogia muutmine (*retopology*).

Lihne pindade muutmine (liigutamine, pööramine, suuruse teisendamine)

Mode: Muutmisrežiim Edit

Hotkey: G/R/S

Menu: Surface » Transform » Grab/Move, Rotate, Scale, ...

Kui sa oled ühe või rohkem kontrollpunkte valinud, saad neid haarata/liigutada (G), pöörata (R) või muuta nende suurust (S) nagu paljude teistegi asjade puhul Blenderis ja nagu seda on kirjeldatud [3D-ruumis manipuleerimise](#) peatükis.

Muutmisrežiimis Edit on sul nende lihtsate muutmiste puhul olemas veel üks lisavõimalus: [proportsionaalne muutmine](#).

Keerulisemad teisendamise tööriistad

Mode: Muutmisrežiim Edit

Menu: Surface » Transform

Teisendamise tööriistu 'kerakujuliseks' To Sphere, 'aja kiiva' Shear, 'mässi' Wrap ja 'lükka/tõmba' Push/Pull on kirjeldatud [võre muutmise](#) peatükis. Pindadel mingeid eraldi teisendamise tööriistu ei ole.

NURBSide kontrollpunktide sätted

Mode: Muutmisrežiim Edit

Panel: Kõvera tööriistad Curve Tools (muutmise kontekst Editing, F9) ja teisendamise omadused Transform Properties


Nagu nägime [eelmisel lehel](#), on NURBSide kontrollpunktidel kaal, mis on selle punkti mõju pinnale. Saad seda määrata kas kõvera tööriistade paneelis Curve Tools asuva suure raskuse määramise nupu Set Weight abil (olles kõigepealt määranud paremale jäävas numbriväljas raskuse) või trükkides väärtuse otse teisendamise omaduste paneelis Transform Properties olevasse numbrivälja W.

Lisamine ja eendamine

Mode: Muutmisrežiim Edit

Hotkey: E (või Ctrl LMB )

Menu: Surface » Extrude



Erinevalt võredest või kõveratest, ei saa sa tavaliselt pinnale otse uusi kontrollpunkte lisada (kombinatsiooniga Ctrl LMB ) , sest ainus viis pinna laiendamiseks on terve U- või V-rea korraga lisamine. Ainuke erand on NURBS-pinna kõver, st pind, milles on igas U- või V-reas ainult üks kontrollpunkt. Sellel erijuhul töötab kõik samuti nagu [kõverate](#) puhul.

Enamasti saab kasutada ainult eendamist. Nagu tavaliselt, toimub pärast selle tööriista käivitamist eendamine koheselt ning sind saadetakse liigutamisrežiimi Grab, mis ootab, et sa uue eendatud pinna selle sihtkohta lohistaksid.

Väga oluline on mõista kahte asja:



- Pinnad on **kahemõõtmelised** objektid – seega ei saa sa eendada mitte midagi pinna sees olevat (nt mõnda “sisemist” rida), see ei oleks loogiline!
- Kontrolliv “ruudustik” *peab jääma “ruudusamaseks”*, mis tähendab, et sa saad eendada ainult tervet rida ning mitte kohti siin ja seal...

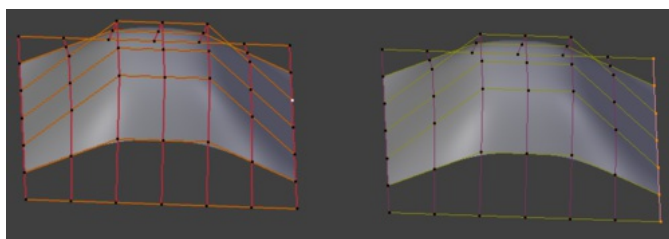
Kokkuvõtlikult võib öelda, et eendamise tööriist (Extrude) töötab siis ja ainult siis, kui valitud on üks terve piirav rida – muidu ei juhtu midagi.

Nagu kõverate puhulgi, ei saa sa oma objektis uut pinda luua mitte millestki lihtsalt ilma midagi valimata kombinatsiooniga Ctrl LMB  klõpsates. Aga erinevalt kõveratest pole olemas “lõikamise” valikut, millega saaksid pinna kaheks osaks eralda, ning seega saad sa uut pinda luua ainult mõnda olemasolevat [kopeerides](#) ( ShiftD) või tekitades uue pinna (lisamise menüü Add...).

Näited

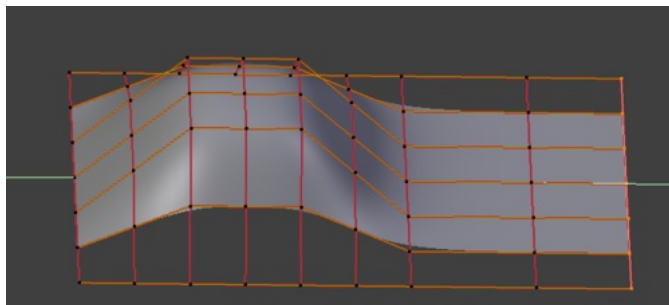
Pildid (*Kontrollpunkt valimine*) kuni (*Valmis*) näitavad tüüpilist pinna külje eendamist.

Piltidel (*Kontrollpunkti valimine*) ja ( ShiftR) valiti kontrollpunktidest piir-rida, valides kõigepealt üh kontrollpunkti nimega “c” ja kasutades seejärel järgnevat kontrollpunktide valimiseks rea valimise tööriista ( ShiftR).



Seejärel serva eendatakse (E), nagu näidatud pildil (*Eendamine*). Pane tähele, kuidas võre on valitud rea kõrval krussi tõmbunud

(mainitud koht on tõstetud helehalli ringikujulise alaga esile). See on nii seetõttu, et ka *uus* eendatud pinna osa on sealsamas kohas.



Liigutades uut osa sellest alast eemale, hakkab pind "lahti jooksuma". Liikumise suund on tähistatud valge noolega "E" ja uus pinna osa tähega "S".

Saad sellist eendamise (või lisamise) protsessi jätkata, kuni jõuad oma mudeli lõpliku kujuni.

Pindade avamine ja sulgemine

Mode: Muutmisrežiim Edit

Hotkey: C

Menu: Surface » Toggle Cyclic

Nagu [kõveraid](#), saab ka pindasid sulgeda (tsükliliseks muuta) ja avada. Kuna pinnad on aga kahemõõtmelised, siis määrad sa seda võimalust U- ja V-telgede puhul eraldi.

Et tsüklilisust ühel teljel sisse või välja lülitada, vajuta klahvi C ja vali [hüpikmenüüst](#) U- (cyclic U) või V-telg (cyclic V). Vastavad pinna välimised servad ühendatakse, et luua "suletud" pind.

Sisemine ja välimine

Pindadel on "sisemine" ja "välimine" külg, millest esimene on must, kuid teine õigesti varjutatud – pindade puhul ei paista olevat "kahepoolset" varjutamise võimalust... Kui sa ühes või teises suunas pinna sulged, võib tulemuseks olla üleni must objekt! Sellisel juhul [vaheta lihtsalt oma pinna "suunda"](#)...

Duplitseerimine (Duplication)

Mode: Muutmisrežiim Edit

Hotkey: ⇧ ShiftD

Menu: Curve » Duplicate

Nagu võrede ja kõverate puhulgi, duplitseerib see käsk valikut. Koopia valitakse nagu tavaliselt ning see on haaramisrežiimis, nii et saad selle teise kohta liigutada.

Kuid pindade puhul on mõned valikud, mida ei saa duplitseerida, millisel juhul viiakse sind koheselt haaramisrežiimi... Reaalselt on kopeeritavad ainult *üksiku lubatava alamruudustiku* moodustavad valikud. Uurime seda lähemalt:

- Sa saad kopeerida üksiku kontrollpunkti. Sellest saad "eendada" mööda U-telge "pinnakõvera" ning seejärel seda unikaalset U-rida mööda V-telge eendada, et luua reaalne uus pind.
- Saad kopeerida ühe pideva rea osa (või loomulikult ka terve rea). Selle tulemuseks on uus **U-rida** ja seda isegi juhul, kui sa valisid (osaliselt) V-rea!
- Saad kopeerida üksikut tervet alamruudustikku.

Pane tähele, et kui sa proovid duplitseerida mitut lubatud "alamruudustikku" (isegi kui need on üksikud punktid) üheaegselt, siis see ei tööta. Sa pead duplitseerimist tegema üksteise järel...

Elementide kustutamine

Mode: Muutmisrežiim Edit

Hotkey: X või Del

Menu: Curve » Delete...

Pinna kustutamise hüpikmenüü Erase pakub sulle kahte valikut:

Selected (valitud)

See kustutab valitud read *ilma* pinda katkestamata (st kahele poole jäävad read ühendatakse omavahel otseselt pärast keskele jäävate ridade kustutamist). Valik peab vastama järgnevatele reeglitele:

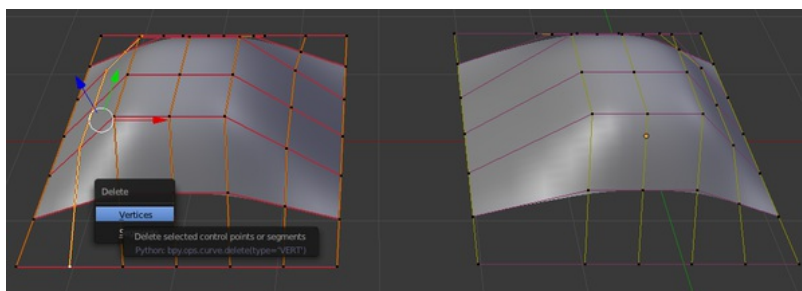
- Valitud võivad olla ainult terved read.
- Valitud read peavad olema samal teljel (st sa ei saa samaaegselt kustutada nii U- kui ka V-ridasid).

Samuti pea meeles, et NURBSide järk ei saa olla kõrgem kui kontrollpunktide arv antud teljel ning seega võib see pärast kontrollpunktide kustutamist väheneda... Loomulikult, kui jääb ainult üks rida, muutub pind "pinnakõveraks", kui jääb ainult üks punkt, pole enam nähtavat pinda, ning kui kustutatakse kõik punktid, kustutatakse ka pind ise.

All (kõik)

Nagu võrede või kõverate puhul, kustutab see valik kõik objektis oleva!

Näide



Enne ja pärast

Pildil (*Enne*) on valitud rida kontrollpunkte, valides kõigepealt punkti nimega "A" ning kasutades seejärel ülejäänud punktide valimiseks kiirkombinatsiooni \diamond ShiftR. Kasutades seejärel [kustutamise menüüd Erase](#) (X), kustutatakse *valitud* kontrollpunktide rida ning tulemuseks on pilt (*Pärast*).

Pindade ühendamine

Mode: Muutmisrežiim Edit

Hotkey: F

Menu: Surface » Make Segment

Nagu [kõverate puhul](#) nõuab pindade ühendamine seda, et oleks valitud üks serv (piiril olev kontrollpunktide rida) kahest eraldi pinnast. See tähendab, et pinnad peavad olema sama objekti osad. Näiteks ei saa sa objektirežiimis Object kahte pinda ühendada – kuid sa saad loomulikult (nagu kõigi teiste objektitüüpide puhul) [ühendada kaks või enam pinnaobjekti](#) üheks objektiks (CtrlJ) – neid aga ei "lingita" ehk ühendata üheks pinnaks... Jah, see on natuke keeruline!

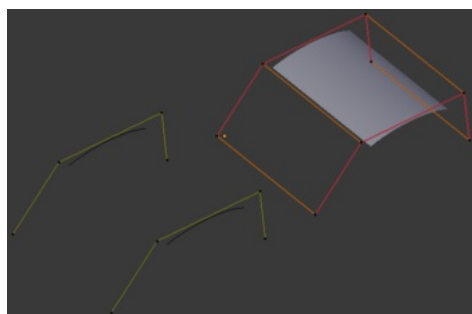
See käsk on võrdne võrede servade või külgede loomisega (siit ka kiirkäsk F) ning seetõttu töötab see ka ainult muutmisrežiimis Edit. Valik tohib sisaldada ainult sama resolutsiooniga (sama arvu kontrollpunktidega) ääreridasid, muidu proovib Blender teha oma parima äranägemise järgi (mida ühendada millega) või kukub ühendamine läbi (kas vaikselt, resolutsioonide mittesobimise veateatega "Resolution doesn't match", kui valitud on erinev arv punkte, või liiga väheste punktide veateatega "Too few selections to merge", kui valisid punkte ainult ühelt pinnalt...)

Seega vali probleemide vältimiseks ainult ääreridasid, millel on sama arv punkte. Pane tähele, et sa saad ühendada ühe pinna U-rea teise pinna V-rea – Blender pöörab automaatselt ühe pinna teljed ringi, et need korrektelt kattuksid.

NURBS-pindade kõveraid kasutatakse sageli erinevate kerede (laevad, kosmoselaevad...) tegemiseks, sest nad määravad terve objekti ulatuses ristlõike ning sa pead neid lihtsalt ilusa, sujuva ja harmoonilise kuju saamiseks "katma", nagu on räägitud ülalpool. Detailsemaid juhiseid [vaata sellest õppetükist](#).

Näited

Vasakpoolne osa pildist (*Valmis ühendamine*) näitab kahte NURBS-pinda (*mitte* NURBS-kõverat) muutmisrežiimis Edit, mis on ühendamiseks valmis. Parempoolne osa pildist on nende kahe kõvera ühendamise tulemus.



Valmis ühendamine.

Tükeldamine (*Subdivision*)

Mode: Muutmisrežiim Edit

Panel: Kõvera tööriistad 1 Curve Tools1 (muutmise kontekst Editing, F9)

Hotkey: W » 1 NumPad

Menu: Surface » Segments » Subdivide, Specials » Subdivide

Pinna tükeldamine on väga lihtne: kasutades kas tükeldamise valikut Subdivide eriomaduste menüüs Specials (W) või kõvera tööriistade 1 paneelis Curve Tools1 olevat tükeldamise nuppu Subdivide, tükeldad sa ühe korra kõik "täielikult valitud ruudustikud", tükeldades iga "nelinurka" neljaks väiksemaks ühikuks.

Kui sa kasutad seda ühemõõtmelise pinna puhul ("pinna kõver"), töötab see tööriist täpselt samamoodi nagu [kõverate oma](#).

Vurr (*Spin*)

Mode: Muutmisrežiim Edit

Panel: Kõvera tööriistad 1 Curve Tools1 (muutmise kontekst Editing, F9)

See tööriist on natuke samane [võre vastavale tööriistale](#) – kuid sellel on vähem valikuid ja võimalusi (tegelikult neid polegi!).

See töötab ainult valitud "pindade" puhul, mis koosnevad ühest *U-reast* (ja ilma ühegi *V-reata*) ehk siis nõnda nimetatud "pinna kõveratega". Tööriist "eendab" seda "ristlõiget" ruudukujulise mustrina ning muudab automaatselt kontrollpunktide kaale, et saada täiuslikult ringikujuline eendus (see tähendab, et pind tuleb mööda V-telge sulgeda). Tegevus järgib täpselt samu printsiipe nagu NURBS-toru (NURBS Tube) ja NURBS-pontšiku (NURBS Donut) puhul.

Muuda suunda (*Switch Direction*)

Mode: Muutmisrežiim Edit

Hotkey: W » 2 NumPad

Menu: Surface » Segments » Switch Direction, Specials » Switch Direction

See käsk pöörab ükskõik millise kõvera, mille puhul on valitud vähemalt üks element, suuna "tagurpidi" (st alguspunkt lõpetab ja *vastupidi*). See on kasulik, kui kasutad kõverat rajana või koos kantimise ja koondamise sätetega.

Teised eriomadused

Mode: Muutmisrežiim Edit

Hotkey: W

Menu: Specials

Eriomaduste menüüs Specials on täpselt samad võimalused nagu [kõverate puhul](#) – kuid raadiuse määramise (Set Radius) ja raadiuse silumisega (Smooth Radius) pole pindade puhul midagi peale hakata...

Konverteerimine

Kuna eksisteerivad ainult NURBS-pinnad, pole olemas "sisemist" konverteerimist.

Kuid eksisteerib "väline" konverteerimine, pinnast võreks, mis töötab ainult objektirežiimis Object. See teisendab pinnaobjekti Surface võreks Mesh, kasutades külgede, servade ja tippude loomiseks pinna resolutsiooni mõlemas suunas.

Topoloogia muutmine (*Retopology*)

Pinnakomponentide nakkumine toimub samamoodi nagu võrede ja kõverate puhul. Lisainfot vaata [topoloogia muutmise](#) peatükist.

Mitmesugune muutmine

Sul on ka mõned samasugused valikud nagu võrede puhul või objektirežiimis Object olles. Sa saad valitud pinna [osadeks jagada](#) (P), muuta teisi valitud objekte ühe või kolme kontrollpunkti [alamateks](#) (CtrlP – pane tähele, et kolmele kontrollpunktile alamaks muutmisel on kõverate puhul imelik tulemus) või [lisada haake \(hooks\)](#), et juhtida kõvera punkte teiste objektide abil.

Kasutada saab ka peegeldamise tööriista Mirror, mis toimib täpselt samamoodi nagu [võre tippude puhul](#).

Tekstid

Mode: Muutmisrežiim Edit (Text)

Panel: Curve and Surface, Font ja Char (Editing kontekst, F9)

Hotkey: F9

Menu: Add » Text



Teksti näited.

Tekstiobjektid (Text) on täpselt need, mida nimi ütleb - need sisaldavad teksti. Neil on olemas ühised elemendid kõverate ja pindadega, kui rääkida kaasaegsetest fondiformaatidest (OpenType, TrueType jne). Need on vektortüüpi ja koosnevad põhiliselt Bézier' kõveratest.

Blenderil on olemas oma fondihaldamise süsteem, et saaks vastavaid tähti ja sümboleid 3D-vaates kuvada. Lisaks sisseehitatud fondile saab kasutada väliseid *PostScript Type 1*, *OpenType* ja *TrueType* fonte. Muide, igat .blend-failis olevat objekti saab kasutada teksti sees oleva tähe/sümbolina.

Blenderi abil saab luua/renderdada 2D- või 3D-teksti ja varjutada seda nii nagu vaja. Lisaks on olemas mitmed küljendamise võimalused nagu joondamine ja tekstikastid. Vaikimisi on tähed seest täis pinnad, täpselt nagu iga suletud 2D-kõver. Loomulikult saab teksti ka eendada (tähtedele paksust anda), samuti kõveraid järgima panna.

Teksti saab ka teist tüüpi objektiks konverteerida (klahviga AltC, objektirežiimis). Konverteerida saab kõverateks või võreks ja nii saab kasutada nende objektitüüpide võimalusi.

Pilt (*Teksti näidised*) näitab mõningate erinevate fontide kasutamist. On näha, et sinine tekst paikneb kõvera peal.

Märkused

Ühes tekstiobjektis võib olla kõige rohkem "50 000" sümbolit. Mida rohkem on tekstiobjektis sümboleid, seda aeglasemalt hakkab arvuti reageerima.

Nagu sa võid objekti- (Object) ja muutmise (Edit) režiimi vahel liikudes näha, fontide (Font) sakk sellest ei muutu. See tähendab, et ühe tekstiobjekti eri osadele ei saa anda erinevaid omadusi, nii nagu seda sai teha võreobjekti puhul. Omaduse muutmine muudab tervet objekti. Nii et font, suurus jne on omane tekstiobjektile tervikuna. On siiski üks võimalus, mille abil saab ühes tekstiobjektis nelja erinevat tekstitüüpi kasutada: nendeks on fondi stiilid "Regular" (tavaline kiri), "Bold" (paks kiri), "Italic" (kaldkiri) ja "Bold&Italic" (paks ja kaldus).

Ressursside optimaalse kasutamise huvides on nii, et mällu loetakse vaid tegelikult kasutuses olevad sümboolid (mitte aga kogu font).

Teksti valimine



Tekst muutmise režiimis Edit.

Muutmise režiimis on näha valget kursorit, mille abil saab määrata, kuhu uus tekst lisatakse - nii nagu suvalises tekstiredaktoris. Kursorit saab liigutada nooleklahvidega (→/↓/←/↑) või Page up/Page down ja ⌘ Home/End klahvidega.

Tekstist mingi osa valimiseks hoia nooleklahvidega liikumise ajal all ⇧ Shift-klahvi. Nii saab määrata materjale, fondi stiili (*normal* / *bold* / *italic*) ja muudki.

Teksti muutmine

Mode: Muutmisrežiim Edit

Hotkey: vaata alt



Tekst muutmisrežiimis.

Teksti muutmine on teiste objektide muutmisest küllaltki erinev ja põhiliselt saab seda teha kahes kohas. Esimene koht on muidugi 3D-vaate aken, kus saab teksti sisestada ja on olemas mõningad klahvikombinatsioonid [stiilide](#) lisamiseks – paljud teiste objektide muutmiseks tuntud klahvikombinatsioonid ei tee muutmisrežiimis tekstiobjekti puhul midagi. Teine koht muutmiseks on objekti andmete (Object Data) sakk. Selle nupu kujutisel on tekstiobjekti puhul F-täht.

3D-vaate päises otseselt tekstiobjektiga seonduvaid nuppe pole. Ka erivalikute menüü (Specials) ei sisalda midagi selle objektiga spetsiifiliselt seonduvat. Pole muutmise, peegeldamise ega muid vahendeid. Teksti saab mõjutada samasuguste töötlejatega (*modifiers*) kui kõveraid.

Teksti muutmine on üsna samasugune nagu tavalistes tekstiredaktorites, aga mitte siiski nii võimalusterohke, lisaks on ka mõningad erinevused:

Muutmisrežiimist väljumine

↔ Tab klahv ei lisa tabulatsioonimärki, vaid lülitab objektirežiimi ja muutmisrežiimi vahel.

Kopeerimine

Teksti lõikepuhvrise kopeerimiseks kasuta klahvikombinatsiooni CtrlC või tööriistariulis (*Tool Shelf*) asuvat **Copy** nuppu.

Lõikamine ja kopeerimine

Teksti lõikepuhvrise kopeerimiseks ja tekstiobjektist eemaldamiseks kasuta klahvikombinatsiooni CtrlX või tööriistariulis olevat olevat **Cut** nuppu.

Asetamine

Lõikepuhvrast teksti asetamiseks on klahvikombinatsioon CtrlV või tööriistariuli nupp **Paste**.

Kogu teksti kustutamine

Kogu teksti kustutamiseks on klahvikombinatsioon Ctrl← Backspace.

Teksti algusesse/lõppu liikumine

↵ Home ja → End klahvid viivad vastavalt rea algusesse või lõppu.

Eelmine/järgmine sõna

Sõnakaupa kursori liigutamiseks kasuta klahvikombinatsioone Ctrl← või Ctrl→.

Tekstipuhver ei suhtle OP-süsteemi puhvriga. See toimib ainult Blenderi sisest. Välise teksti sisestamist vaata allpool olevast osast [teksti sisestamine](#).

Teksti sisestamine

Teksti sisestamiseks on olemas kolm varianti: kirjutamine, lõikepuhvrast asetamine või tekstifaili sissetoomine.

Tekstifaili saab sisse tuua manüüst Text » Paste File. Avaneb failisirvija aken (File Browser) ja selle abil saab sisse tuua sobiva UTF-8 faili. Vaata jällegi, et selles poleks liiga palju tähti, et programmi tagasiside ei muutuks liialt aeglaseks.












Šriftid (fondid)

Mode: Muutmisrežiim Edit

Panel: Font (Editing (muutmise) kontekst, F9)

Fondi seadete (Font) paneelil on teksti väljanägemise muutmiseks palju valikuid.

Šriftifailide (fontide) laadimine ja muutmine

	18thCtnit.ttf	29 KB
	18thCtrKurStart.ttf	89 KB
	18thCtrKurTxt.ttf	84 KB
	1610_Cancellaresca_lim.TTF	116 KB
	BIRTH OF A HERO.ttf	116 KB
	CRBLATRIAL.otf	45 KB
	CRBLATRIAL.ttf	54 KB
	CRIALTRIAL.otf	59 KB
	CRIALTRIAL.ttf	76 KB
	kabog.ttf	23 KB
	kingthingspetrock.ttf	14 KB

Type 1 fondifaili laadimine.

Vaikimisi näitab Blender kõiki nelja valikut oma sisseehitatud fondiga Bfont. Sisseehitatud font on alati saadaval ja selle nimeks kuvatakse Bfont. Esimesest ikoonist avaneb rippmenüü, kus on näha Blenderis hetkel olemasolevad fondid. Iga stiili jaoks saab valida ühe fondi.

Teistsuguse fondi laadimiseks vajuta laadimise nupule ja navigeeri sobiva fondifailini. Failisirvija File Browser näitab ainult neid faile,

mis on laadimiseks sobivad, nagu on näha joonisel (*Type 1 fondifaili laadimine*). Mittesobivaid faile kuvatakse vaid siis, kui failisirvija aknas on välja lülitatud lehtrikujuline nupp.

Unix note

Fondifailid asuvad tavaliselt `/usr/lib/fonts`, mõned ka `/usr/lib/X11/fonts`, kuid see ei ole alati nii. Neid võib olla ka sellistes kohtades nagu `/usr/share/local` või `/usr/local/share` ja nad võivad paikneda ka alamkataloogides.

Kui sa valid sellise fondifaili, mida Blender ei tunnista, ilmub veateade "Not a valid font".

Jäta meelde, et uut fonti kasutatakse kõigi seda kasutavate stiilide juures. Iga stiili jaoks saab aga kasutada eraldi fondifaili. Näiteks kaldkirja saamiseks tuleb laadida "Italics"-tüüpi font. Pärast fondifaili laadimist saab seda rakendada tervele tekstiobjektile. Kokku saab ühes tekstiobjektis kasutada nelja erinevat stiili (**Normal**, **Italics**, **Bold**, **Bold-Italics**).

Blender ise ei vaata, et sinu poolt mingi stiili jaoks üles laaditud font ka selle stiili kirjeldusele vastaks, st et kaldkiri ka tõesti kaldus oleks. Selle abil saab lihtsalt nelja erinevat tüüpi teksti kasutada - igaühe jaoks saab valida erineva fondifaili või selle sees oleva stiili.

Objektid kui fondid

Sa võid "luua" omaenda fondi otse Blenderis! See on võrdlemisi keeruline tegevus, nii et kirjeldame seda detailiselt:

- Esiteks on vaja luua tähed/sümbolid. Iga üksik täht võib olla suvalist tüüpi objekt (võre, kõver, meta vms). Kõik peavad olema nimetatud järgmiste reeglite kohaselt: prefiks ja selle järel tähe nimi (näiteks "ft.a", "ft.b" jne.).
- Järgmisena tuleb tekstiobjektile sisse lülitada duplitipud (*Dupli Verts*). Kui tekstiobjekt on aktiivne, saab selle kätte objekti andmete saki alt Duplication sektsioonist.
- Nüüd tuleb objekti andmete saki alt Font sektsioonis Object Font väljale kirjutada fondiobjekti nimi. Praeguse näite korral siis "ft".

Ja nüüd, kui vastava prefiksi ja tähelaiendiga objekt on olemas, kuvatakse vastava tähe asemel omavalmistatud kujundit. "Nähtavaks jääb ka originaaltäht". Täheobjekti valmistamisel tuleb teha nii, et selle "keskpunkt" paikneks objekti vasemas alumises servas.

Fontide formaatimine

Mode: Muutmisrežiim Edit

Panel: Font (Editing (muutmise) kontekst, F9)

Paneelil Font on terve rida tüpograafiat ja küljendust muutvaid valikuid.

Size (suurus)

Tekstibloki suurus (üksikute tähtede suurust eraldi muuta ei saa). Siiski on võimalik erinev suurus valida neljale võimalikule erinevale stiilile.

Shear (kalle)

Kirja kaldenurk. "See ei ole tegelikult kaldkiri", kuna kaldega saab kalde väärtust muuta.

Rasvane, kald- ja allajoonitud kiri



Pane linnuke kasti, et näidata näiteks rasvast kirja

Italics (kaldkiri)

Pärast sissetrukkimist saab seda Italic-nupuga sisse ja välja lülitada. Nupp **Italic** on olemas ka tööriistariiulis (Tool Shelf).

Bold (paks kiri)

Nupp Bold lülitab sisse ja välja paksu kirja stiili. Nupp **Bold** on olemas ka tööriistariiulis.

Bold and Italics (paks ja kaldus)

See on kahe eelneva kooslus, kuid siin võib kasutuses olla ka teistsugune font.

Underline (alla joonitud)

Lülitab sisse teksti allajoonimise. Ka see nupp on tööriistariiulis olemas.

Underline position (allajoonimise positsioon)

Selle abil saab allajoonimise kriipsu vertikaalselt ja horisontaalselt nihutada.

Underline height (allajoonimise paksus)

Sellega määratakse kriipsu paksus.

Blenderi Bold ja Italic nupud ei tööta analoogselt teiste programmidega. Need hakkavad toimima siis, kui vastav font on seal väljal ära määratud. Selle kohta vaata [Ülaltpoolt](#).

Bold/Italics/Underline stiili saamiseks võid selle juba enne trükkimist sisse lülitada. Kuid nende saamiseks võid ka esmalt teksti sisse trükkida, soovitud ala ära märkida ja siis vastavast nupust stiili ära muuta.



Paks kiri.

Pildil (*Paks kiri*) on üks font kasutuses "Te" jaoks ja teine font "xt" jaoks.

Suurtähtede ja väiketähtede kasutus

Tööriistariulis olevate nuppude **To Upper** ja **To Lower** abil saad valitud tähed vastavalt suurtähtedeks või väiketähtedeks muuta.

Väikeste suurtähtede kasutamiseks lülita sisse Small Caps.

Nende suurust saad muuta Small Caps Scale seadistuste alt. Pane tähele, et sama suurus Small Caps Scale kehtib kõigile tähtedele, millele on määratud Small Caps formateering.

Joondamine

Paragraph paneelil on valikud teksti joondamise ja paigutamise jaoks.



Lõigu sakk

Joonda tekst (*Align*)

Left (vasakule)

Joondab teksti tekstikasti vasemasse serva. Kasti mitteolemasolu korral on joondamiseks tekstiobjekti "keskpunkt".

Center (keskele)

Kasutab tekstikasti keskele joondamist. Kasti puudumise korral on joondamiseks tekstiobjekti "keskpunkt".

Right (paremale)

Joondab teksti tekstikasti paremasse serva. Kasti puudumise korral on joondamiseks tekstiobjekti "keskpunkt" ja tekst "kasvab" trükkides vasemale poole.

Justify (joonda mõlemasse serva)

Muudab sõnade vahel olevate tühikute laiust, nii et tekst oleks nii vasemalt kui paremalt äärest vastu raami. Sõna sees olevad tähevahed aga ei muutu.

Flush (sõrenda mõlemasse serva)

Muudab sõnade vahel olevate tühikute laiust ja sõna sees olevate tähtede vahekaugust, nii et tekst oleks nii vasemalt kui ka paremalt äärest vastu raami: n-ö. venitab mõned sõnad laiemaks.

Nii Justify kui ka Flush toimivad ainult raami sees olevana.

Tähevahed (*Spacing*)

Character (tähemärk)

Kui palju suurendada-vähendada tähtede omavahelist kaugust

Word (sõna)

Sõnade vahel olevate tühikute laiuse määramine. Seda saab teha ka klahvikombinatsioonidega Alt← ja Alt→, mis suurendavad/vähendavad vastavalt **0.1** võrra.

Line (rida)

Reavahe muutmine.

Nihe (*Offset*)

X offset ja Y offset

Selle abil saab teksti X- ja Y-teljel normaalasendi suhtes nihutada. Raamide kasutamisel mõjutab see kogu tekstiobjekti teksti.

Erimärgid (*Special Characters*)

Mode: Muutmisrežiim Edit

Menu: Text » Special Characters

Mõningad erimärgid saab kätte Alt-klahviga või Text-menüüst 3D-vaate päises.

Siin on nendest tähtedest kokkuvõte:

AltC: Copyright (©)	AltR: Registreeritud kaubamärk (®)
AltG: Kraad (°)	AltX: Korrutamise märk (×)
AltS: Saksa kahekordne "ss" (ß)	AltF: Valuuta märk (₣)
AltL: Inglise nael (£)	AltY: Jaapani jeen (¥)
Alt1: Ülakerjas 1 (¹)	Alt2: Ülakerjas 2 (²)
Alt3: Ülakerjas 3 (³)	Alt.: Rõngas
Alt?: Hispaania küsimärk (¿)	Alt!: Hispaania hüüumärk (¡)
Alt<: Vasempoolne nurklik jutumärk («)	Alt>: Parempoolne nurklik jutumärk (»)

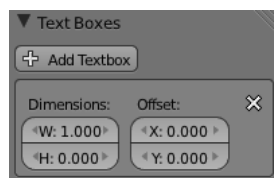
Kõik klaviatuurilt trükitavad tähed peaksid toimima. Mitmete erimärkide saamiseks (mida klaviatuuril pole) saab kasutada klahvikombinatsioone. Näiteks "täpitähtede" saamiseks trüki esmalt sisse tavaline täppideta täht, vajuta Alt← Backspace ja seejärel trüki sisse soovitud "täpid". Siin on mõned näited:

A, Alt← Backspace, ~:	ä A, Alt← Backspace, `:	á A, Alt← Backspace, `:	à
A, Alt← Backspace, O:	å E, Alt← Backspace, `:	ë O, Alt← Backspace, /:	ø

Teksti raamid (*Text Boxes*)

Mode: Object or Edit modes

Panel: Font (Editing (muutmise) kontekst, F9)



Teksti raam.

Raami abil saab teksti raami sisse jaotada. Ühe tekstiobjekti kohta võib olla vabalt valitud hulk raame. Kõigi tekstikastide asukohta ja suurust saab vabalt muuta.

Tekst voolab sõnade kaupa väiksema numbriga tekstikastist suurema suunas. Tekst liigub edasi juhul, kui ühte tekstiraami tekst enam ära ei mahu. Viimase tekstiraami korral läheb mitte äramahtuv tekst kastist välja.

Tekstiraamid toimivad üsna sarnaselt küljendusprogrammidega, näiteks Scribus. Tekstiraamide abil saab kontrollida teksti küljendust.

Raami omadusi saab muuta paneelilt Text Boxes.

Tekstiraami suurus (*Frame size*)

Iga uue raami tekitamisel on selle laius ja kõrgus null ning seda pole veel näha.

Nullsuurusega tekstiraame ignoreeritakse ja tekst hakkab nendest kohe välja voolama. Isegi sel juhul, kui on olemas järgmine nullist suuremate mõõtudega raam.

Raami nähtavaks muutmiseks tuleb määrata selle suuruseks suurem arv kui null.

Märkus

Tehniliselt võttes pole kõrgus kunagi **0.0**, sest font ise annab kõrguse.



Raami laius

Joonisel (*Raami laius*) on tekstiobjekt, mille laius on **5.0**. Kuna see on suurem kui **0.0**, on see nüüd näha ja seda kuvatakse katkendjoonega. Tekst on kastist väljunud, kuna see on jõudnud viimase raamini, mis on praegusel juhul vaikimisi lisatud raam.

Raami lisamine/kustutamine

Raami lisamiseks vajuta nuppu **Add Textbox**, mis asub paneelil Text Boxes. Uue raami lisamisel on see sama suur ja samas kohas nagu viimane raam. Uue raami korral muuda X ja/või Y välja. Ainult X-i muutmine tekitab uue veeru.

Raami kustutamiseks vajuta nuppu **Delete**. Kõigi järgnevate raamide tekst voolab nüüd vastavalt uuele olukorrale uuesti.

Näidis: Teksti voolamine



Tekst 1.

Kahe või enama tekstiraamiga saab teksti paremini paigutada. Tekita näiteks tekstiobjekt ja kirjuta sinna "Blender on üllilahe"; vaata (*Tekst 1*). Tekstiobjektil on olemas küll raam, aga see pole veel nähtav, sest raami laius (Width) on **0.0**.

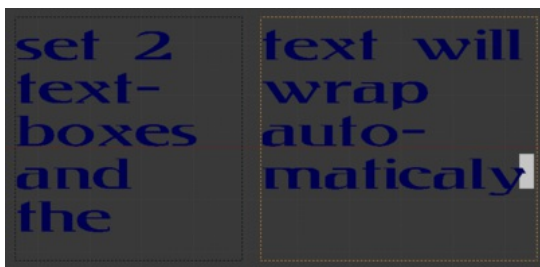


Tekst 2.

Pane laiuse väärtuseks **5.0**. Nüüd on juba raami laius näha ja uelt realt jätkamine toimub vastavalt sellele, nagu näha pildil (*Tekst 2*). Nagu näha, voolab tekst üle raami serva. Nii on see sellepärast, et tekst on jõudnud viimase raamini, mis praegusel juhul on vaikimisi esimene raam.

Uue raami lisamisel ja sellele laiuse-kõrguse andmisel voolab tekst sinna edasi.

Näide: mitme veeru kasutamine



Tekst 5.

Mitme veeru kasutamiseks tuleb esimesele raamile anda kõrgus ja laius ja siis uus raam tekitada. Uue raami suurus on sama mis eelmisel raamil. Määra X asukohaks midagi suuremat või väiksemat, kui see on eelmisel raamil. Vaata pilti (*Tekst 5*).

Mitme materjali kasutamine

Mode: Muutmisrežiim Edit

Panel: Link and Materials (Editing (muutmise) kontekst, F9)

Iga täht võib olla erinevast materjalist.

Materjali võib ära määrata enne trükkimist, kuid soovitud teksti võib ära märkida ka pärast trükkimist ja sellele seejärel materjalide saki all **Assign**-nupuga vastav materjal määrata.

Näited



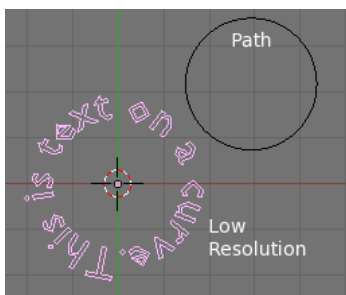
Red Green Blue.

Näite (*Red Green Blue*) tegemiseks on vaja esmalt kolm materjali tekitada. Igale sõnale saab materjali määrata, kui see esmalt ära märkida ja siis **Assign**-nuppu vajutada. (*Red Green Blue*) on jätkuvalt üks ja terviklik tekstiobjekt.

Tekst kõveral

[Kõveratöötlejaga](#) saad teksti panna kõvera peale jooksma.

Näide



Tekst kõveral.

Pildil (*Tekst kõveral*) näed sa 2D Bézier'i kõveral olevat teksti.

Kõveratöötleja lisamiseks tuleb tekstiobjekt esmalt konverteerida võreks, kasutades AltC ja valides *Mesh*.

Märkus

On olemas ka tekst kõveral (*Text on Curve*) võimalus, kuid kõveratöötleja pakub rohkem võimalusi.

Kõvera kuvamise valikud

Mode: Object or Edit modes

Panel: Curve and Surface (Editing (muutmise) kontekst, F9)

Nagu näha paneelilt Curve and Surface, on tekstil ja kõveral palju samasuguseid seadistusi.

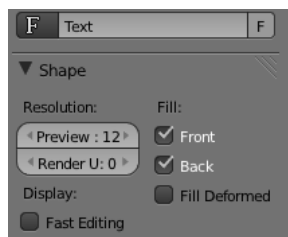
Resolutsioon (*Resolution*)

Preview (eelvaade)

[Resolutsioon](#) vaateaknas.

Render (renderdus)

[Resolutsioon](#) renderdamisel.



Kuju sätteid

Fast Editing (kiire muutmine)

Lülitab muutmisrežimis kõverate täitmise välja.

Täide (*Fill*)

Selle abil on ära määratud, kuidas on tähekeha sisu täidetud siis, kui Geometry paneelil on sisse lülitatud eendamine (Extrude) või kantimine (Bevel).

Front (ees)

Täidab pinnaga esipoole.

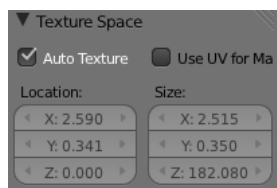
Back (taga)

Täidab pinnaga tagapoole.

Fill Deformed (täida deformeeritud)

Täidab kõverad pärast võtmevormide (*shape keys*) ja töötlejate (*modifiers*) rakendamist.

Tekstuudid



Tekstuuri sätteid

Use UV for Mapping (kasuta UV-laotust)

Kasuta tekstuuri paigutamisel UV-laotust.

Auto Texture Space (automaatne tekstuuriruum)

Kohandab aktiivse objekti teisendamisel automaatselt selle tekstuuri.

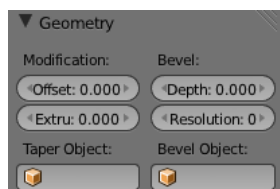
Kantimine ja eendamine

Tekstiobjektidega on täpselt samasugused võimalused nagu kõveratega. Vaata [Kõveratest eendamise võimalused](#).

Muutmine (*Modification*)

Offset (nihe)

Määrab ära, kui kaugel asub kõver sees- või väljaspool oma algsest asukohtast. See on n-ö. täispumpamise/tühjaksimise efekt.



Teksti geomeetria sätted

Extrude (eenda)

Määrab eendamise ulatuse. Eendamise profiili saab määrata sobiva kõvera abil väljalt Taper Object. Vaata [Kõvera eendamine](#)

Kantimine (*Bevel*)

Teksti kantimisega saab selle servadele lisada servi ümardavad kandid.

Depth (sügavus)

Määrab kandi ulatuse

Resolution (resolutsioon)

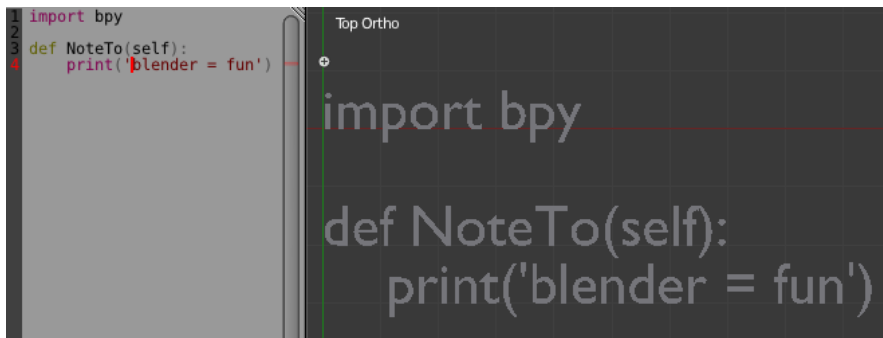
Määrab kandi sujuvuse. Väärtus "0" lisab täiesti lameda kandi. Suurem number muudab kandi sujuvamaks. Kõvera abil kandi profiili (Bevel Object) loomisel määratakse kandi ulatus ja ümarus kõvera abil.

Konverdi tekst tekstiobjektiks



Lihne moodus Blenderisse teksti saamiseks on selle trükkimine tekstiredaktorisse ning seejärel selle konverteerimine tekstiobjektiks.

Selleks lae tekstiredaktorisse tekstifail või trüki sinna midagi. Seejärel konverteeri see lihtsaks tekstiobjektiks või muuda iga rida eraldi objektiks. Need võimalused leiad *Edit* saki alt.



vasakul tavaline tekst, paremal tekstiobjektiks muudetud tekst.

Meta objektid

Mode: Objekti režiim Object või muutmisrežiim Edit

Hotkey: ⇧ ShiftA

Menu: Add » Meta

Meta objektid on *kaudsed pinnad*, mis tähendab, et nad *ei ole otseselt* tippude (nagu võred) või kontrollpunktidega (nagu pinnad) defineeritud: nad eksisteerivad *protseduuriliselt*. Meta objektid on otsesõnu matemaatilised valemid, mida Blender arvutab jooksu pealt.

Metadele väga iseloomulik visuaalne omadus on see, et nad on vedelad *elavhõbedalikud* või *savilaadsed* vormid, mille kuju on “ümar”. Ja mis veelgi olulisem, kui kaks meta objekti teineteise lähedale satuvad, hakkavad nad omavahel suhtlema. Nad “segunevad” või “ühinevad” nagu veetilgad ja eriti nagu veetilgad kaaluta olekus (mistõttu nad on muideks väga kasulikud veejугade modelleerimisel, kui sa ei soovi tegeleda vedeliku simulatsiooniga). Kui nad pärastpoole teineteisest eemale liiguvad, taastavad nad oma algse kuju.

Igaühte neist määrab eraldi [matemaatiline baasstruktuur](#) ning sa saad iga hetk neid aktiivse elemendi paneeli Active Element kaudu omavahel vahetada.

Tavaliselt kasutatakse Meta objekte eriefektide jaoks või modelleerimise alusena. Sa võid näiteks kasutada metade kogumit mudeli algse kuju loomiseks ning konverteerida selle siis edasiseks modelleerimiseks mõneks teiseks objektitüübiks (no tegelikult saad valida ainult võresid). Meta objektid on ka kiirtejälitusega renderdamise puhul väga efektiivsed.

Pane tähele, et Meta objektid käituvad objektirežiimis Object natuke teistmoodi, kui seda on kirjeldatud [allpool](#).

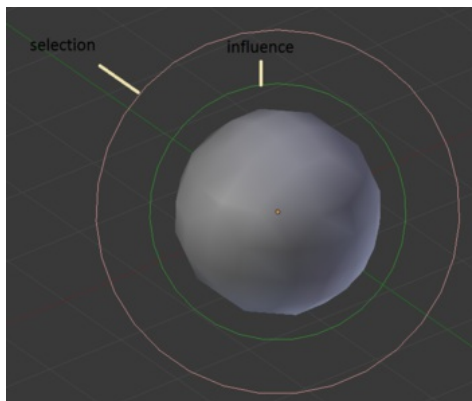
Primitiivid

Meta lisamise alammenüüs Add » Meta on viis meta “primitiivide” (ehk konfiguratsioonide) eelvalikut:

- Metapall (Meta Ball) lisab meta, mille alusstruktuuriks on punkt.
- Metatoru (Meta Tube) lisab meta, mille alusstruktuuriks on joonsegment.
- Metatasapind (Meta Plane) lisab meta, mille alusstruktuur on tasapinnaline.
- Metaellipsoid (Meta Ellipsoid) lisab meta, mille alusstruktuuriks on ellipsoid.
- Metakuup (Meta Cube) lisab meta, mille alusstruktuuriks on volumeetriline kuup.

Kuvamine

Objektirežiimis Object näidatakse valmisarvutatud võret koos musta “valikuringiga” (valimisel muutub roosaks). Et saada rohkem teada metadest objektirežiimis Object, loe [allpoolt](#).

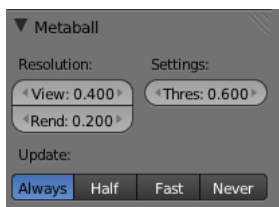


Metapalli (Meta Ball) näidis.

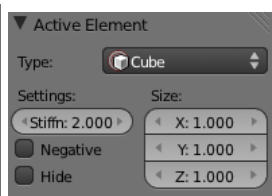
Muutmisrežiimis Edit (*Metapalli (Meta Ball) näidis*) kuvatakse meta võrena (varjutatud või must traatvõre, kuid loomulikult ilma tippudeta) koos kahe värvilise ringiga: punane valimiseks (valituna roosa) ja roheline meta jäikuse otseseks määramiseks (loe [allpoolt](#) – valituna heleroheheline). Pane tähele, et kui sa just ei tegele parajasti suuruse teisendamisega (S), on aktiveeritud roheline ring samaväärne punase aktiveeritud ringiga.

Metapalli sätted

Kõik stseenis olevad meta objektid suhtlevad omavahel. Metapalli (MetaBall) osas olevad sätted kehtivad kõigi meta objektide puhul. Muutmisrežiimis Edit ilmub aktiivse elemendi paneel Active Element, mille kaudu saad muuta üksikuid metaelemente.



Globaalsed meta sätted.



Individaalsed meta sätted.

Resolutsioon

Resolutsioon (Resolution) määrab meta objekti poolt genereeritava võre resolutsiooni.

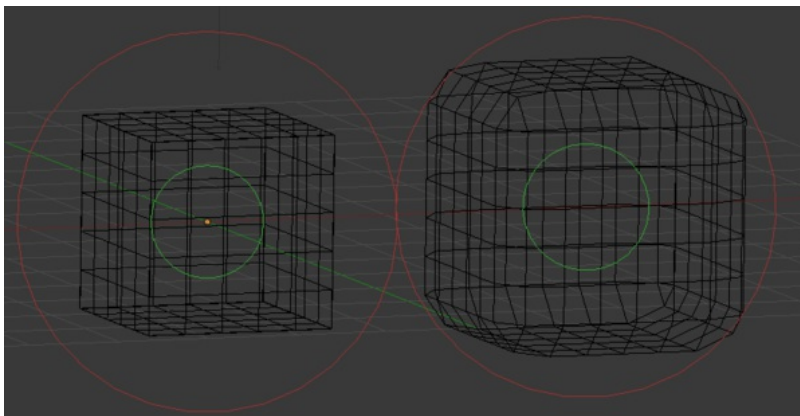
View (vaade)

Genereeritud võre resolutsioon 3D-vaates. Vahemik on **0.05** (peenim) kuni **1.0** (jämedaim).

Render (renderdus)

Genereeritud võre resolutsioon renderdamisel. Vahemik on **0.05** (peenim) kuni **1.0** (jämedaim).

Üks moodus aluseks oleva matemaatilise struktuuri nägemiseks on vähendada resolutsiooni (Resolution), suurendada läve (Threshold) ja määrata jäikus (Stiffness, loe altpoolt) tsipakene lävest kõrgemaks. Pildil (*Alusstruktuur*) on metakuup, millel on järgnevad sätete väärtused: Resolution **0.410**, Threshold **5.0** ja Stiffness veidi kõrgem ehk **5.01**.



Vasakul: Alusstruktuur, paremal: Kuju.

Näed selgelt aluseks olevat kuubi struktuuri, mis annab metakuubile kuju.

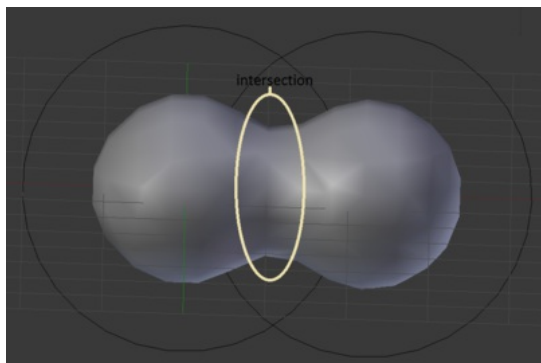
Lävi (mõju)

Mode: Objekti režiim Object või muutmisrežiim Edit

Panel: Metapall MetaBall (muutmise kontekst Editing, F9)

Lävi (Threshold) määrab, palju meta pind "mõjutab" teisi metasid. See määrab *välja taseme*, mille juures pinda arvutatakse. See säte on metaobjektide [grupi](#) puhul globaalne. Mida kõrgem on lävi, seda suurem on iga meta poolt teistele avaldatav mõju.

Mõju on kahte tüüpi: **positiivne** või **negatiivne**. Tüüpi saab muuta muutmisrežiimis Edit aktiivse elemendi paneelis Active Element oleva negatiivsuse nupuga Negative. **Positiivset** võid endale ette kujutada tõmbumisenä ja **negatiivset** tõukumisenä. Negatiivne meta lükkab positiivsete meta objektide võred endast eemale.



Positiivne.

Positiivsed mõju määratletakse tõmbumisenä, mis tähendab, et võred sirutuvad üksteise poole, kui nende *mõjuringid* ristuvad. Pilt (*Positiivne*) näitab kahe *positiivse* mõjuga metapalli *mõjuringi* ristumas.

Pane tähele, kuidas võred on üksteise poole tõmbunud. Valgega ümbritsetud ala näitab rohelisi *mõju* ringe ristumas.

Värskendamine

Metade teisendamisel (haara/liiguta, muuda suurust jne) on sul neli kuvamise "režiimi", mis asuvad metapalli paneeli MetaBall värskendamisnuppude grupis Update:

- Always (alati) – kuva teisendamise ajal metat täisresolutsiooniga.
- Half Res (poolresolutsioon) – kuva teisendamise ajal metat poole selle traadimahu Wiresize resolutsiooniga.
- Fast (kiiresti) – ära näita teisendamise ajal meta võret.
- Never (mitte kunagi) – ära näita meta võret kunagi (see pole eriti soovituslik, kuna selle puhul on meta nähtav ainult renderdamise ajal!).

Need peaksid aitama, kui sul tekib raskusi (metad on päris arvutamismahukad), kuid kaasaegsete arvutite puhul ei tohiks seda juhtuda, kui sa just ei kasuta palju metasad või väga kõrget resolutsiooni.

Meta struktuur

Tehnilised detailid

Ametlikum metaobjekti definitsioon võiks olla *juhataav struktuur*, mida võib mõista staatilise välja allikana. See väli võib olla kas positiivne või negatiivne ning seeläbi naabruses olevate juhatavate struktuuride välja kas tõmmatakse või tõugatakse.

Kaudset pinda saab defineerida pinnana, kus kõigi juhatavate struktuuride kolmemõõtmelised väljad omandavad teatud väärtuse. Näiteks metapall, mille juhataav struktuur on punkt, loob enda ümber isotroopse (st kõigis suunades identse) välja ning selliste objektide pinnad on konstantse välja väärtuse puhul kerad, mille keskmes on juhataav punkt.

Metaobjektid on lihtsalt matemaatilised valemid, mis suhtlevad üksteisega loogiliste tehete kaudu (ja ehk AND, või ehk OR) ning mida saab üksteisele liita või üksteisest lahutada. Selle meetodi nimi on ka **konstruktiivne tahke geomeetria** (*Constructive Solid Geometry*, CSG). Lähtuvalt oma matemaatilisest olemusest kasutab CSG vähe salvestusmälu, kuid nõuab arvutamiseks palju protsessori võimsust.

Alusstruktuur

Panel: Metapalli tööriistad MetaBall tools (muutmise kontekst Editing, F9), teisendamise omadused Transform Properties

Blenderis on viis meta tüüpi, millest igaühel on oma alusstruktuur (ehk juhataav struktuur). Muutisrežiimis Edit saad sa seda struktuuri muuta, kasutades kas metapalli tööriistade paneelis MetaBall tools olevaid vastavaid nuppe või teisenduse sätete paneelis Transform Properties (N) olevat ripploendit. Olenevalt struktuurist võib olla lisaparametreid, mis asuvad nii Transform Properties kui ka MetaBall tools paneelides.

Pall (Ball, punkt, nulldimensiooniline struktuur)

See on kõige lihtsam meta, millel puuduvad lisasätted. Kuna see on lihtsalt punkt, loob see isotroopse välja, mille tulemuseks on kerajas pind (seetõttu kutsutaksegi seda Blenderis metapalliks (Meta Ball) või palliks (Ball)).

Toru (Tube, sirge joon, ühemõõtmeline struktuur)

See on meta, mille pinna loob valitud pikkusega sirge joone poolt tekitatav väli. Tulemuseks on silinderjas pind, mille suletud otsad on ümarad. Sellel on üks lisaparameter:

- dx: joone pikkus (ja seega ka toru pikkus – vaikeväärtus on **1.0**).

Tasapind (Plane, nelinurkne tasapind, kahemõõtmeline struktuur)

See on meta, mille pinna loob nelinurkse tasapinna poolt tekitatav väli. Tulemuseks on kindla paksuse ja ümarate servadega rööptahukakujuline pind. Sellel on kaks lisaparametrit:

- dx: nelinurga pikkus (vaikeväärtus on **1.0**).
- dy: nelinurga laius (vaikeväärtus on **1.0**).

Pea meeles, et vaikimisi on see tasapind ruut.

Ellipsoid (ellipsoidne maht, kolmemõõtmeline struktuur)

See on meta, mille pinna loob ellipsoidse mahu poolt tekitatav väli. Tulemuseks on ellipsoidne pind. Sellel on kolm lisaparametrit:

- dx: ellipsoidi pikkus (vaikeväärtus on **1.0**).
- dy: ellipsoidi laius (vaikeväärtus on **1.0**).
- dz: ellipsoidi kõrgus (vaikeväärtus on **1.0**).

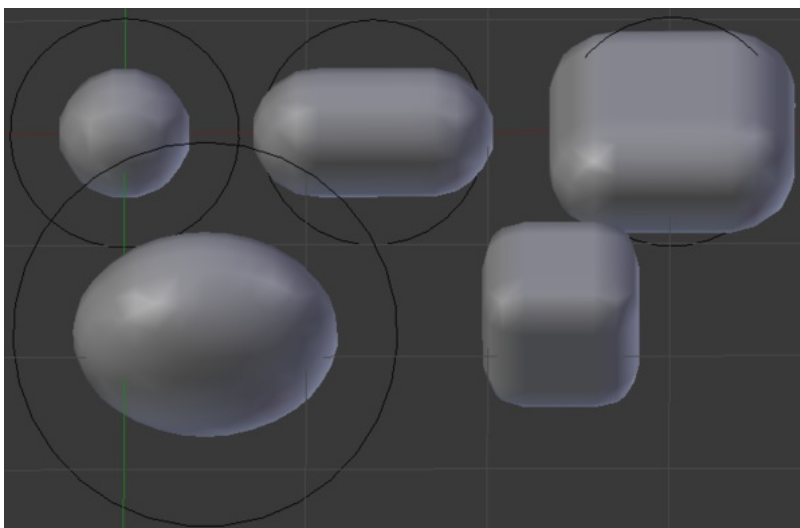
Pane tähele, et vaikimisi on see maht kerakujuline, mis tekitab palliga samase keraja meta.

Kuup (Cube, risttahukakujuline maht, kolmemõõtmeline struktuur)

See on meta, mille pinna loob risttahukakujulise mahu poolt tekitatav väli. Tulemus on risttahukakujuline pind, millel on ümarad servad. Nagu sa võisid arvata, on sellel kolm lisaparametrit:

- dx: risttahuka pikkus (vaikeväärtus on **1.0**).
- dy: risttahuka laius (vaikeväärtus on **1.0**).
- dz: risttahuka kõrgus (vaikeväärtus on **1.0**).

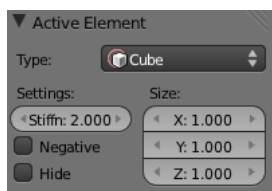
Pane tähele, et vaikimisi on maht kuubikujuline.



Viis meta primitiivi.

Metaobjektide muutmise

Muutmisrežiimis ilmub aktiivse elemendi paneel Active Element. Need sätted mõjutavad ainult valitud meta-elementi.



Aktiivse elemendi paneel.

Meta kuju

Tüübi menüü Type lubab sul muuta metaobjekti kuju, nagu seletati eelnevalt.

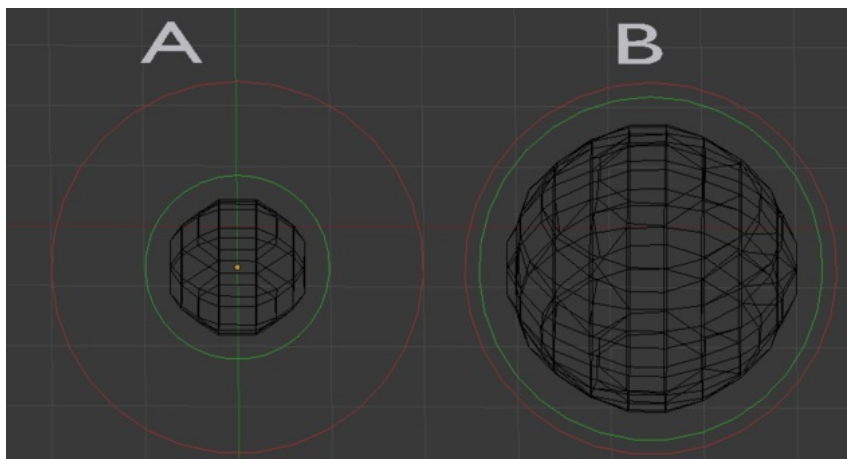
Jäikus (*Stiffness*)

Koos läve sättega Threshold mõjutab jäikus (Stiffness) objekti mõju piirkonda. Kuigi lävi on sama kõigile ühes objektis (või isegi samas [objekti perekonnas](#)) olevatele metadele, on jäikus iga meta puhul erinev.

Sisemise rohelise ringi mõõtkava muutmise muudab jäikuse väärtust (Stiffness). Jäikus määrab kui palju metaobjektist täidetakse. See määrab põhimõtteliselt siis selle, kui tundlik on meta teiste metade poolt tulevatele mõjutustele. Madalama jäikuse puhul hakkab meta moonduma juba kaugemalt. Kõrgem väärtus tähendab, et meta peab teisele metale olema lähemal, et asuda kokku sulama.

Kui metaobjekt jõuab teise meta "piirkonda" (*range*), hakkavad need omavahel suhtlema. Nad ei pea päriselt üksteisega kattuma ning sõltuvalt läve (Threshold) ja jäikuse (Stiffness) sätetest ei ole seda üldiselt ka vaja. Jäikuse (Stiffness) muudab näitlikuks *roheline ring*.

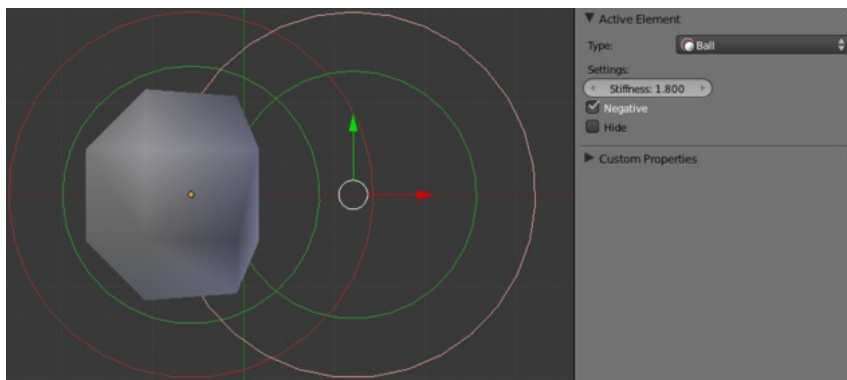
Vahemik on **0.0** kuni **10.0**. Kuid selleks, et olla nähtav, peab jäikuse (Stiffness) väärtus olema kõrgem kui läve (Threshold) oma. Saad jäikuse ringi ka visuaalselt muuta, valides selle hiireklahviga RMB ning aktiveerides kiirvalikuga S mõõtkava teisendamise režiimi.



Jäikus.

Pildil (*Jäikus*) on metapallil nimega "A" väiksem jäikuse (Stiffness) väärtus kui pallil nimega "B". Nagu näed, on nende *rohelistest ringidest* raadius erinev.

Negatiivne mõju (*Negative Influence*)



Negatiivne

Positiivse mõju vastandefektiks oleks *negatiivne* mõju: objektid tõukavad teineteist eemale. Pilt (*Negatiivne*) näitab metapalli ja metatasapinda, milledest esimene on negatiivne ja teine positiivne. Pane tähele, kuidas negatiivne meta pole nähtav: näha on ainult seda ümbritsevat ringi. Sedasi viitabki Blender sellele, et objekt on negatiivne.

Kera liigutamine tasapinnani paneb tasapinna võre "iseendasse tõmbuma" ehk sissepoole kokku kukkuma. Kui sa liigutad tasapinna kerast eemale, taastab tasapinna võre ennast tagasi.

Et muuta meta "negatiivseks", vali see muutmisrežiimis ja pane aktiivse elemendi paneeli *Active Element* negatiivse kasti *Negative* linnuke.

Elementide peitmine

Nagu objektirežiimis Object, saad sa ka valitud meta(sid) peita ja siis uuest nähtavaks teha. See on väga kasulik, kui soovid oma vaadet natuke puhastada... Pane tähele, et muutmisrežiimis Edit jäävad alati alles kaks punast ja rohelist ringi ning objektirežiimis Object jääb alles valimise ring.

Praeguse valiku peitmiseks kasuta kiirkäsku H, metapallide tööriistade MetaBall tools all olevat peitmise nuppu Hide või menüüvalikut Metaball » Hide MetaElems » Hide Selected.

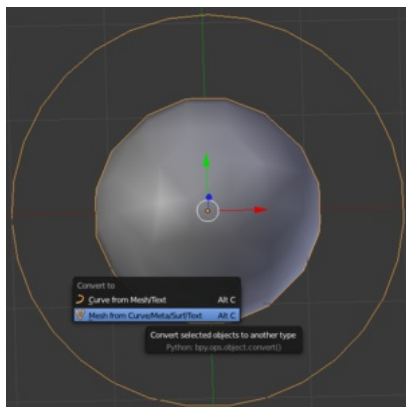
Et peita kõike peale selle, mis on parajasti valitud, kasuta kiirkäsku ⇧ ShiftH või menüükirjet Metaball » Hide MetaElems » Hide Deselected.

Peidetu näitamiseks kasuta kiirkäsku AltH või sobivat kirjet menüüs Metaball » Hide MetaElems. Samuti võid metapalli tööriistade paneeli MetaBall tools peitmise nupu Hide välja lülitada.

Elementide kustutamine

Metade puhul kustutamise menüüd Erase ei ole. Ilmub lihtsalt hüpikaken, mis küsib, kas soovid valitud metad kustutada. Lihtne!

Konverteerimine (*Conversion*)



Konverteerimise menüü

Metasid saab konverteerida võredeks, kuid sul on võimalus ka algne metaobjekt alles jätta (st luua "reaalse" konverteerimise asemel uus võre...). Pea meeles, et uue võre resolutsiooniks on traadimaht (Wiresize), mitte renderdusmaht (Rendersize).

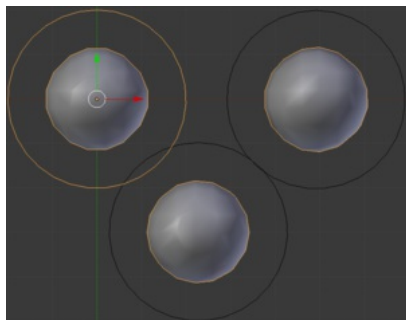
Metade konverteerimiseks vajuta objektirežiimis AltC ja vali võre (*Mesh*).

Objektide perekonnad

Metaobjektidel on objektirežiimis Object teistsugune käitumine kui teistel objektitüüpidel – neid saab uuesti nõndanimetatud "perekondadesse" grupeerida.

"Perekond" on viis mitme metaobjekti uuesti grupeerimiseks, mille tulemuseks on midagi väga sarnast mitme meta samas objektis hoidmisele.

Perekonna määrab objekti nime vasakpoolne osa (enne punkti). Pea meeles, et objekti nimi on enamikus paneelides objekti väljal "OB" ja **mitte** väljal "MB", mis tähistab meta andmebloki nime... Näiteks *perekonna* osa nimest "MetaPind.001" on "MetaPind". Iga samas "perekonnas" olev meta on teistega allpool toodud viisidel seotud.



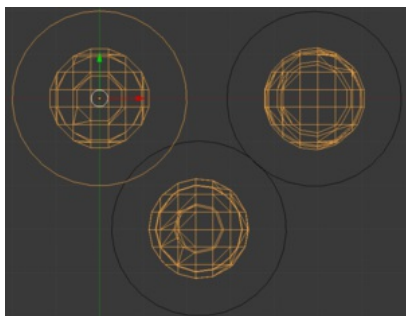
Metapalli baas.

Metade perekondi kontrollib üks *baas*-metaobjekt, mille määrab objekti nimi ilma parema pooleta. Kui meil on näiteks viis metat nimedega "MetaAsi", "MetaAsi.001", "MetaAsi.002", "MetaAsi.003" ja "MetaAsi.004", oleks *baas*-metaks "MetaAsi".

Baas-metaobjekt määrab baasi, resolutsiooni, läve ja teisendused. Sellel on ka materjali ja tekstuuri ala. *Baas*-meta on teiste grupis olevate metade ülem (või "omanik", st teised metad oleksid justkui selle "sees" või sellega seotud).

Näited

Pilt (*Metapalli baas*) näitab *baas*-metat sildiga "B". Teised kaks metaobjekti on *alamad*. Alamate valikuringid on alati mustad ja grupi võre oranž. Kuna metad on grupeeritud, moodustavad nad ühtse võre, mida saab alati valida, valides ükskõik millise grupis oleva meta võre. Näiteks pildil (*Metapalli baas*) on valitud ainult alumine kera (ülem), kuid nagu sa näed, on esiletõstetud nii ülema võre *kui* ka kõigi alamate võred.



"Baasi" mõõtkava teisendamine.

Baas-metaobjekt määrab grupi **polügonideks jagunemise** (võre struktuuri) ja seetõttu määrab ka kõigi alammetade (*mitte-baas*) polügonideks jagunemise. Kui me *baas*-metat muudame, muutub ka alamate polügonideks jagunemine. Kui me aga muudame alamaid, jääb polügonideks jagunemine samaks.

Vihjed

"Polügonideks jagunemise" arutamine *ei tähenda*, et erinevad võred üksteise suunas või üksteisest eemale ei moonduks (metaobjektid mõjutavad teineteist alati tavapärasel moel, hoolimata sellest, kas nad on sama perekonna liikmed või mitte). See tähendab hoopis, et võre baasstruktuur muutub ainult siis, kui muudetakse *baas*-objekti. Kui sa näiteks teisendad *baas*-objekti mõõtkava, siis muutub ka alamate võrestruktuur. Pildil (*"Baasi" mõõtkava teisendamine*) on *baas*-meta mõõtkava vähendatud, mille tulemusena on vähenenud ka kõik alamate võrestruktuuri mõõtkava. Nagu sa näed, on alamate võre resolutsioon suurenenud, kuid *baasi* oma vähenenud. *Alamate suurus ei muutunud!*

Tühiobjektid (*Empties*)

Empty on tühiobjekt. Seda ei renderdata ja muutmisrežiimi sellel objektil ei ole. Küll aga saab objektirežiimis seda teisendada (nihutada, pöörata, suurendada-vähendada) ja muuta interaktiivses 3D-vaates kuvatavat indikaatoritelgede suurst. Tühiobjektid on tähtsad ja kasulikud objektid. Mõned näited tühiobjektide kasutamisest:

- Objektide grupi ülemobjektina
- Piirajate (*constraints*) sihtmärgina
- Kohatäitjana (*placeholder*)
- Taageldamisel (*rigging*) kontrolliva objektina
- Fookuskauguse asukoha määrajana (DOF distance)
- Objektide reastamisel reatöötajas vahekauguse määrajana
- Modelleerimisel aluseks võetud piltide kuvamiseks

Kuvamise seaded

Plain Axes (ainult teljed)

Ainult telgjoonte kuvamine. 3 ühepikkust sirget, mis on X-, Y- ja Z-telgede sihis.

Arrows (nooled)

3 noolt, mis on X-, Y- ja Z-telgede positiivses suunas. Iga noole juures kuvatakse ka vastava telje tähis.

Single Arrow (üksik nool)

Ainult üks joon Z-telje positiivses suunas.

Circle (ring)

Ringina kuvamine XZ-telgede tasandil.

Cube (kuup)

Võrestikrežiimis kuubikuna kuvamine

Sphere (kera)

Kolme rõnga abil kujutatav kerasamane objekt

Cone (koonus)

Y-telje positiivses suunas suunatud koonus.

Image (pilt)

Uutes versioonides võib tühiobjekti abil kuvada ka pilte. Selle abil võib kuvada modelleerimisel aluseks olevaid pilte. See on alternatiiv 3D-vaate taustapildi määramisele (*View/Properties/Background Images*). Tühiobjekti abil kuvatakse pilt alati, sõltumata vaatenurgast. 3D-vaate taustapildi puhul aga on konkreetne pilt nähtav vaid konkreetse vaate korral. Muus osas on seadistused samad, nagu kirjeldatud [taustapiltide seadete peatükis](#).

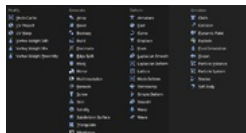
Size (suurus)

Tühiobjekti kuvamise suurus interaktiivses 3D-vaates. See ei muuda tühiobjekti enda suurst. Muutub vaid kuvatava ikooni suurus.

Modifiers - töötlejad

Mode: Kõik režiimid

Panel: Modifiers (Töötlejad)



Töötlejate menüü

Töötlejad on automaatsed teisendused, mis muudavad objekti väljanägemist selle lähtekuju muutmata. Töötlejate abil on võimalik teha automaatselt mitmeid teisendusi ja efekte, mis käsitsi tehes nõuaks palju nüri tööd (näiteks pinnatükeldus (*subdivision surface*)), säilitades sealjuures objekti algse topoloogia. Töötlejad muudavad objekti väljanägemist, renderdamist, kuid mitte tema geometriat. Ühele objektile on võimalik korraga lisada mitmeid töötlejaid, mis moodustavad [töötlejate pinu \(Modifier Stack\)](#). Kui soovitakse töötlejate efektid muuta püsivaks (st muuta algse objekti kuju vastavaks töötleja väljundile), saab töötlejad rakendada nupuga Apply.

On olemas kolme tüüpi töötlejaid:

Moodustajad

Generate-gruppi kuuluvad töötlejad kas muudavad olemasoleva objekti geometriat või lisavad objekti võrele automaatselt uusi geomeetrilisi detaile.

[Rida](#) (Array)

loob baasvõrest rea sellele sarnanevaid (korduvaid) kujusid.

[Kantija](#) (Bevel)

kandib valitud võre servi ja tippe.

[Kahendttöötleja](#) (Boolean)

kombineerib/lahutab/leiab ühisosa kahe võre vahel.

[Ehitaja](#) (Build)

koostab animatsiooni, milles võre luuakse samm-sammult.

[Lihtsustaja](#) (Decimate)

vähendab võre külgede arvu.

[Servalõhestaja](#) (Edge Split)

muudab võre servade varjutuse osaliselt teravaks.

[Mask](#) (Mask)

võimaldab varjata võre osasid.

[Peegeldaja](#) (Mirror)

loob sümmeetrilise võre, peegeldades objekti mööda koordinaattelgi. Sellega väheneb töö objekti modelleerimisel ja muutmisel ainult poole, neljandiku või kaheksandikuni esialgsest.

[Paralleelvõre](#) (Multiresolution)

võimaldab vooldada võret erinevatel detailsustasetel.

[Kruvi](#) (Screw)

loob lihtsast kahemõõtmelisest objektist heeliksilaadse keha. Samanab võre muutmise kontekstis olevale Screw (kruvi) tööriistale.

[Paksuse lisaja](#) (Solidify)

annab võre külgedele (*face*) paksuse.

[Pinnatükeldaja](#) (Subdivision Surface)

loob interpoleeritud, esialgsest pinnast siledama pinna.

[UV projitseerija](#) (UV Project)

projitseerib võrele UV-koordinaadid.

[Nahk](#) (Skin)

genereerib automaatselt topoloogia

Moonutajad

Moonutamise (Deform) grupi töötlejad muudavad ainult objekti kuju. Neid saab rakendada võrele ning sageli ka tekstidele, kõveratele, pindadele ja/või sõrestikele.

[Skelett](#) (Armature)

kasutab luude süsteemi objekti poseerimiseks ja animeerimiseks.

[Lähendaja](#) (Cast)

lähendab võre, pinna või sõrestiku kuju sfäärile, silindrile või risttahukale.

[Kõver](#) (Curve)

painutab objekti olemasoleva kõvera järgi.

[Nihutaja](#) (Displace)

deformeerib objekti tekstuuri järgi.

[Haak](#) (Hook)

lisab tippudele või juhtpunktidele haagi, mis võimaldab neid distantstilt deformeerida.

[Sõrestik](#) (Lattice)

võimaldab objekti deformeerida sõrestiku abil.

[Võremoonutaja](#) (Mesh Deform)

võimaldab ühte objekti deformeerida, muutes teise võre (deformatsiooniümbriku) kuju (samaset sõrestikuga deformeerimisele).

[Sobitaja](#) (Shrinkwrap)

võimaldab sobitada objekti teise objekti pinnale (mähkida teise objekti ümber).

[Moonutaja](#) (Simple Deform)

lisab objektile mõned spetsiifilised deformatsioonid.

[Siluja](#) (Smooth)

silub võre geomeetriat. Samaneb võre silumise (Smooth) tööriistale võre muutmise kontekstis.

[Väänaja](#) (Warp)

venitab sõrestikku kahe määratud punkti vahel.

[Laine](#) (Wave)

deformeerib objekti, tekitades (animeeritud) lained.

Simulatsioonid

Simulatsioonide (Simulate) grupi töötlejad käivitavad simulatsioonid. Enamasti lisatakse need töötlejad pinusse automaatselt, kui osakeste süsteem (Particle System) või vastav füüsikasimulatsioon (Physics Simulation) objektile sisse lülitatakse ja nende ainus roll on määrata koht töötlejate pinus, kus vastavaid simulatsioone rakendatakse. Üldiselt on nende töötlejate seadistused kättesaadavad eraldi paneelidel.

[Kangas](#) (Cloth)

simuleerib kanga omadusi. See töötleja lisatakse pinusse, kui võre märgitakse kangaks.

[Kokkupõrge](#) (Collision)

simuleerib objektide vahelist kokkupõrget.

[Plahvatus](#) (Explode)

paneab objekti osakeste süsteemi abil plahvatama.

[Vedelik](#) (Fluid)

objekt on osa vedelikusimulatsioonist. See töötleja lisatakse, kui võre märgitakse vedelikuks.

[Osake](#) (Particle Instance)

paneab objekti käituma samaselt osakesele, kasutades objekti võre kuju.

[Osakeste süsteem](#) (Particle System)

tähistab pinus osakeste süsteemi, seega lisatakse see töötleja automaatselt alati, kui objektile liidetakse osakeste süsteem.

[Suits](#) (Smoke)

simuleerib realistlikku suitsu.

[Pehme keha](#) (Soft Body)

pehmete ja elastsete kehade füüsika. Lisatakse pinusse, kui objekt märgitakse pehme kehana.

[Dünaamiline maalimine](#) (Dynamic Paint)

võimaldab kasutada objekti või osakeste süsteemi teise objekti materjali maalimiseks.

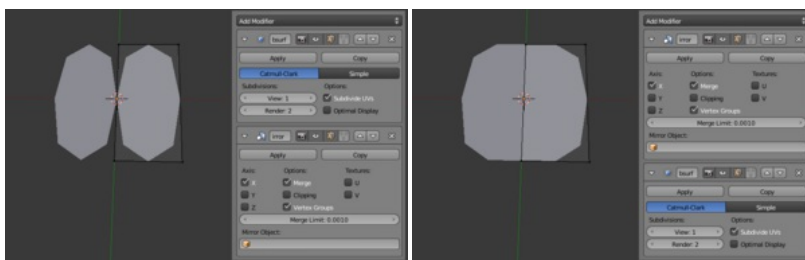
[Ookean](#) (Ocean)

genereerib realistliku, animeeritud merepinna.

Töötleva

Töötleva on defineeritud kui objektile rakendatud "protsess" või "algoritm". Neid saab objektidele rakendada interaktiivselt, ilma esialgselt objekti jäädavalt muutmata, enam-vähem suvalises kasutaja valitud järjekorras. Sedalaadi funktsionaalsust nimetatakse "töötlevate pinuks" (*modifier stack*) ja kasutatakse paljudes 3D-programmides.

Töötlevad lisatakse objektile menüüst Modifiers (töötlevad). Mõned tööriistad ja skriptid, näiteks Decimate (lihtsustaja) ja Solidify (paksuse lisaja) on uues Blenderi versioonis muudetud töötlevateks. Töötlevate pinus mõjutab tulemust järjekord, milles töötlevaid rakendatakse. Õnneks saab töötlevaid lihtsalt ümber paigutada, vajutades üles ja alla suunatud nooltega nuppe. Näiteks pildil (*Pinu järjekord*) on näha [pinnatükeldaja \(SubSurf\)](#) ja [peegeldaja \(Mirror\)](#), mis on pinus vastupidistel kohtadel.



Pinu järjekord

Esimeses näites on Mirror (*peegeldaja*) pinus viimasel kohal. Tulemus näib koosnevat kahest osast. Teises näites on peegeldaja pinus esimesel kohal ja tulemus on ühtne, kokkuliidetud pind.

Nagu näha, erineb tulemus eelmisest näitest. See tähendab, et töötlevate järjekord pinus on lõpptulemuse määramisel oluline.

Kasutajaliides



Paneeli ülevaade
(Pinnatükeldaja Subsurf)
näitel)

Iga töötleva pärineb algselt erinevast kohast Blenderis, seetõttu on igaühel oma unikaalsed valikud ja erilised nõudmised. Siiski on kõikide töötlevate kasutajaliidesel samad põhielemendid - vaata (*Töötleva paneel (Pinnatükeldaja Subsurf näitel)*).

Üleval servas on paneeli päis. Iga ikoon märgib erinevat töötleva võimalust (vasakult paremale):

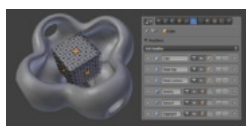
1. Nool — Paki paneel kokku ja näita ainult päist.
2. Selle töötleva ikoon ning väli töötleva nimega — vaikimisi on seal töötleva enda nimi. See on kõigi sama tüüpi töötlevate seas unikaalne.
3. Kaamera — Näita töötleva efekti renderdamisel.
4. Silm — Näita töötleva efekti 3D-aknas.
5. Kast — Näita töötleva efekti muutmise režiimis (Edit Mode). Sõltuvalt töötleva tüübist võib see nupp ka puududa.
6. Kolmnurk — Rakendab töötleva muutmise režiimis (Edit Mode) muutmise võrele (näitab tulemust muutmise ajal). See ikoon käivitab Cage Mode (muutmise puuri režiimi).
7. Nool üles — Liigutab töötlevat pinus ülespoole.
8. Nool alla — Liigutab töötlevat pinus allapoole.
9. Rist — Eemaldab töötleva pinust.

Päise all on kaks nuppu:

1. Apply (rakenda) — Rakendab selle töötleva efekti püsivalt lähteobjektile.
2. Copy (kopeeri) — Loob töötlevast koopia pinu algusesse.

Nende nuppude all on alampaneel konkreetse töötleva valikutega.

Pinu



Selles näites on lihtne tükeldatud kuup muudetud töötlevate pinu abil üpris kompleksseks objektiks. ([blend](#))

Töötleva lisamisel asetad sa ta *pinusse*. Kui töötlevad on kord lisatud (uued töötlevad liiguvad alati pinu põhja), on neid võimalik suvaliselt ümber korraldada.

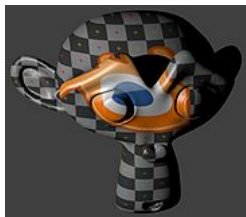
Mõned töötlevad sobivad ainult kindlat tüüpi objektidega. Seda vihjatakse filtreerides Add Modifier (lisa töötleva) nuppu töötlevate paneelil Modifiers. Sellelt avanevas menüüs näidatakse ainult neid töötlevaid, mida antud tüüpi objektile lisada saab. Mesh (võre)

tüüpi objektidele on võimalik lisada kõiki tüüpe töötlejaid, samas kui näiteks sõrestikele (Lattice) saab lisada ainult üksikuid.

UV Project Modifier - UV-projektseerija

Mode: Kõik režiimid

Panel: Modifiers (töötledjad) (Generate (loomine))

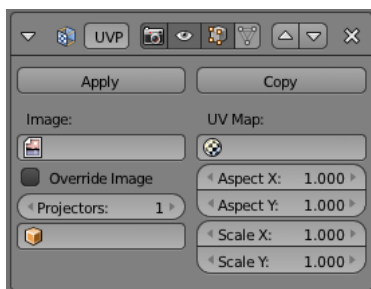


Blenderi logo projektsioon Suzanne'ile.

UV-projektseerija töötab nagu projektor. Ta projekteerib mööda kuni kümne objekti -Z-telge UV-kaardi ja kirjutab selle baasobjektile nii, nagu valgus seda "tabab". Soovi korral võib see töötledja objekti pinnatekstuuri üle kirjutada.

Laadi näitefail

Valikud



UV layer (UV-kiht)

Millist UV-kihti muuta. Vaikimisi kasutatakse parasjagu aktiivset kihti.

Image (pilt)

Selle töötledja projitseeritav pilt. Seda ei pea sisestama; sa võid lihtsalt genereerida UV-koordinaadid mujal kasutamiseks.

Override Image (asenda pilt) allpool määrab, kuidas seda pilti kasutatakse.

Override Image (asenda pilt)

- Kui see on sisse lülitatud, asendatakse võre kõigi tippude pinnatekstuur selle pildiga. See muudab pildi korduvaks, mis ei ole tavaliselt soovitud efekt.
- Kui see on välja lülitatud, mõjub töötledja ainult neile tippudele, mille pinnatekstuuriks on see pilt.

Projectors (projektorid)

<Objects> (objektid)

On võimalik sisestada kuni 10 projektorobjekti. Iga külje jaoks kasutatakse seda objekti, mis on kõige paremini joondatud tema pinnanormaaliga

Projektsioon toimub piki -Z telge (st kaamerast või lambist otse alla).

Kui projektoriks on kaamera, järgib projektsioon selle perspektiivi (või ortograafilise projektsiooni) seadistusi.

Aspect X/Y (X/Y suhe)

Scale X/Y (X/Y skaala)

Need võimaldavad pilti pisut sättida.

Kasutamine

UV-projektseerija on kasulik selleks, et muuta kohtvalgused varieeruvamaks ja samuti teha kleepse ([decal](#)), et stseeni korduvust vähendada.

Töötledja pildivalikut tavaliselt ei kasutata: selle asemel lisatakse objekti materjalile tekstuur, mis seotakse selle UV-kihiga, mida töötledja muudab. See võimaldab vältida tekstuuri kordumist, kui määrata Texture » Image Mapping » Extension (Tekstuur -> Pildilaotus -> Laiendus) väärtuseks *Clip* (kärbi).

Tipugrupi kaalude töötleja (*WeightVGroup Modifiers*)

Mode: Kõik režiimid

Panel: Modifiers (Modifiers omadused)

Kirjeldus

Tipugrupi kaalude töötlejad (*WeightVGroup modifiers*) töötavad mõjutatava objekti tipugruppidega, muutes selle kaale ja/või seda, millised tipud gruppi kuuluvad.

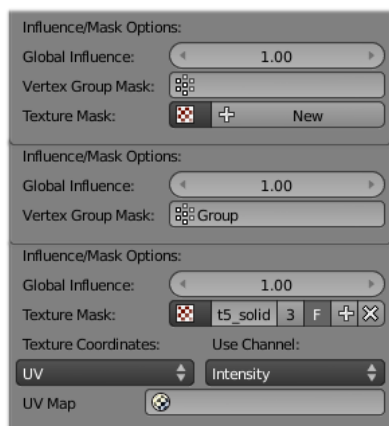


Need töötlejad piiravad kaalude väärtusi rangelt standardses vahemikus $[0.0, 1.0]$. Seega lähevad kaduma kõik väärtused, mis on alla **0.0** üle **1.0**!

Praegu on olemas kolm tipugrupi kaalude töötlejad:

- [Tipu kaalude muutmine](#).
- [Tipu kaalude segamine](#).
- [Tipu kaalude lähedus](#).

Üldised sätted



Tipugrupi kaalude töötlejate mõju/maskimise osa (*influence/masking*).

Kolm tipugrupi kaalude töötlejat jagavad mõningaid sätteid, mis määravad nende mõju mõjutatavale tipugruppile.

Global Influence (globaalne mõju)

Töötleja üldine mõju (**0.0** jätab tipugrupi kaalud puutumata, **1.0** on tavaline mõju).



Pane tähele, et mõju kehtib ainult kaaludele – tippude lisamine/eemaldamine tipugrupist sellele ei allu ning seega väärtus **0.0** seda ära ei hoia!

Vertex Group Mask (tipugrupi mask)

Lisa tipugrupp, mille kaalud korrutatakse iga tipu puhul eelnevalt globaalse mõju väärtusega. Kui mõni tipp ei ole maskiva tipugrupi osa, on selle maskimise kaal (ja seega selle mõju) null.

Texture (tekstuur)

Lisa tekstuur, mille iga tipu väärtused korrutatakse eelnevalt globaalse mõju väärtusega. Saad valida, millist tekstuurikanalit väärtustena kasutatakse.

See on tavapärane tekstuuri ID väli. Kui see on määratud, ilmuvad uued sätted:

Texture Coordinates (tekstuurikoordinaadid)

Kuidas tekstuur on võrele laotatud... Sul on neli valikut:

- Local (lokaalne): kasuta lokaalseid tippude koordinaate.
- Global (globaalne): kasuta tippude koordinaate globaalses ruumis.
- Object (objekt): kasuta tippude koordinaate mõne teise objekti ruumis.
- UV: kasuta UV-kihi koordinaate.

Use Channel (kasuta kanalit)

Millist kanalit kaalu faktori allikana kasutada (intensiivsus, RGB, HSV, alfa – ma usun, et need valikud räägivad enda eest ise).

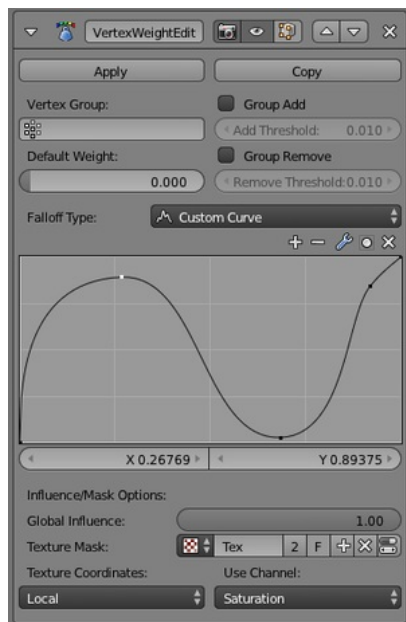
Object (objekt)

Objekt, mida objekti laotuse Object puhul kasutada

UV Layer (UV-kiht)

UV-kiht, mida UV laotuse puhul kasutada

Tipu kaalude muutmise töötleja (*Vertex Weight Edit Modifier*)



Tipu kaalude muutmise töötleja
WeightVGEEdit paneel.

See töötleja on mõeldud ühe tipugrupi kaalude muutmiseks.

Iga tipu puhul on protsess järgnev:

1. [Valikuline] See tekitab laotuse, kas vastavalt mõnele eeldefineeritud funktsioonile või spetsiaalsele laotuskõverale.
2. See lisab mõju faktori ning vajadusel ka tipugrupi või teksturi maski (nullväärtused tähendavad algset kaalu, **1.0** tähistab täielikult laotatud kaalu).
3. See määrab tipule saadud kaalu ja/või saab vastavalt vajadusele tipu grupist eemaldada, kui selle kaal on allpool määratud läve, või selle lisada, kui selle kaal on üle vastava läve.

Sätted

Vertex Group (tipugrupp)
Mõjutatav tipugrupp.

Default Weight (vaikekaal)
Vaikekaal, mis määratakse kõigile tippudele, mis ei ole antud tipugrupis.

Falloff Type (hajumise tüüp)
Laotuse tüüp:

- Linear (lineaarne) – Laotust ei toimu.
- Custom Curve (spetsiaalne kõver) – Lülitab sisse kõveraga laotamise. Selle peale ilmuvad kõvera sätted.
- Sharp (terav), Smooth (sujuv), Root (ruut) ja Sphere (kera) on klassikalised laotamise funktsioonid sakilisimast ümaraimani.
- Random (juhuslik) – Muudab kaalud täiesti juhuslikuks!
- Median Step (mediaani samm) – Loob binaarsed kaalud (**0.0** või **1.0**), mille lõikamise väärtuseks (*cutting value*) on **0.5**.

Group Add (lisa gruppi)
Lisab sellesse tipugruppi need tipud, mille lõplik kaal ületab lisamise läve Add Threshold.

Group Remove (eemalda grupist)
Eemaldab sellest tipugrupist tipud, mille lõplik kaal on väiksem kui eemaldamise lävi Rem Threshold.

Tipu kaalude segamise töötleja (*Vertex Weight Mix Modifier*)



Tipu kaalude segamise töötleja
WeightVGMix paneel.

See töötleja segab mõjutatava tipugrupi mõne teise tipugrupiga (või tavalise väärtusega), kasutades erinevaid operatsioone.

Samuti on sellel säte, mis lubab valida, milliste tippudega tegeletakse (kõik, ainult esimese või teise tipugrupi osad jne).



See tähendab, et see töötleja *võib* lisada mõjutatavasse tipugruppi tippe (see ei eemalda samas tippe kunagi). Rohkem räägitakse sellest allpool.

Sätted

Vertex Group A (tipgrupp A)

Mõjutatav tipugrupp.

Default Weight A (vaikekaal A)

Vaikekaal, mis määratakse kõigile tippudele, mis ei ole antud tipugrupis.

Vertex Group B (tipugrupp B)

Teine tipugrupp, millega mõjutatavat segatakse. Jäta see tühjaks, kui soovid sisse segada tavalise väärtuse.

Default Weight B (vaikekaal B)

Vaikekaal, mis määratakse kõigile tippudele, mis ei ole teises tipugrupis.

Mix Mode (segamise režiim)

Kuidas mõjutavad esimese tipugrupi tippe teise tipugrupi tipud. Sul on seitse valikut:

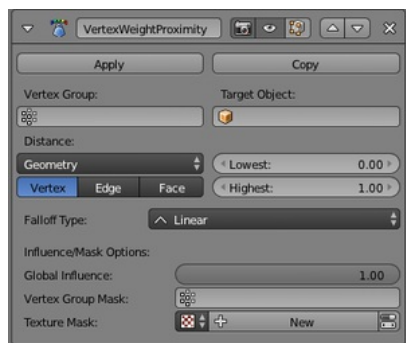
- Replace weights (asenda kaalud) asendab lihtsalt mõjutatavad kaalud teiste kaaludega.
- Add to weights (liida kaaludele) liidab mõlemad väärtused kokku.
- Subtract from weights (lahuta kaaludest) lahutab teise grupi kaalud esimese omadest.
- Multiply weights (korruta kaale) korrutab mõlemad kaalud.
- Divide weights (jaga kaale) jagab esimese grupi kaalud teise grupi kaaludega.
- Difference (erinevus) arvutab mõjutatavate kaalude ja teise grupi kaalude erinevuse (see on lihtsalt lahutamise absoluutne väärtus).
- Average (keskmine) leiab mõlema kaalu keskmise väärtuse.

Mix Set (segamise komplekt)

Milliste tippudega töötada. Sul on viis valikut:

- All vertices (kõik tipud) mõjutab kõiki tippe ning ei piirdu tipugrupi sisuga. *See valik võib mõjutatavasse tipugruppi tippe lisada.*
- Vertices from group A (A-grupi tipud) mõjutab ainult mõjutatavasse tipugruppi kuuluvaid tippe.
- Vertices from group B (B-grupi tipud) mõjutab ainult teise grupi kuuluvaid tippe. *See valik võib mõjutatavasse tipugruppi tippe lisada.*
- Vertices from one group (ühe grupi tipud) mõjutab ainult tippe, mis kuuluvad vähemalt ühte kahest tipugrupist. *See valik võib mõjutatavasse tipugruppi tippe lisada.*
- Vertices from both groups (mõlema grupi tipud) mõjutab ainult tippe, mis kuuluvad mõlemasse tipugruppi.

Tipu kaalude läheduse töötleja (*Vertex Weight Proximity Modifier*)



Tipu kaalude läheduse töötleja WeightVGProximity paneel.

See töötleja määrab valitud tipugrupi kaalud vastavalt objektile (või selle tippude) ja teise sihtobjektile (või selle geomeetria) vahelisele kaugusele.

Sätted

Vertex Group (tipugrupp)

Mõjutatav tipugrupp.

Target Object (sihtobjekt)

Objekt, mille järgi kaugusi arvutada.

Läheduse režiim (*Proximity mode*)

- Object Distance (objekti kaugus) kasutab töödeldava võreobjekti ja sihtobjekti vahelist kaugust kõigi mõjutatava tipugrupi tippude kaaluna.
- Geometry Distance (geomeetria kaugus) kasutab iga tipu ja sihtobjekti või selle geomeetria vahelist kaugust.

Geomeetria kauguse režiimil on kolm lisasätet, mis võimaldavad kasutada sihtobjekti keskpunkt asukoha asemel selle geomeetria (kui lülitad neist sisse rohkem kui ühe, siis kasutatakse lühimat arvutatud kaugust). Kui sihtobjektile puudub geomeetria (see on näiteks

tühiobjekt või kaamera), siis kasutatakse vaikimisi objekti kauguse režiimi Object Distance.

Vertex (tipp)

See määrab iga tipu kaalu vastavalt selle kaugusele sihtobjekti lähimast tipust.

Edge (serv)

See määrab iga tipu kaalu vastavalt selle kaugusele sihtobjekti lähimast servast.

Face (külj)

See määrab iga tipu kaalu vastavalt selle kaugusele sihtobjekti lähimast küljest.

Lowest Dist (väikseim kaugus)

Kaugus, mille puhul määratakse kaaluks **0.0**. See võib olla suurem kui suurim kaugus Highest Dist, et luua vastupidiseid laotamise efekte.

Highest Dist (suurim kaugus)

Kaugus, mille puhul määratakse kaaluks **1.0**. See võib olla väiksem kui väikseim kaugus Lowest Dist, et luua vastupidiseid laotamise efekte.

Falloff Type (hajumise tüüp)

Mõned eeldefineeritud laotamise funktsioonid. Vaata [ülalolevat tipu kaalude muutmise töötleja osa](#).

Näited

Sihtobjekti kauguse kasutamine

Esimese näitena muudame dünaamiliselt laine töötletat Wave vastavalt töötletajale allutatud tipugruppile.

Lisa ruudustiku võre Grid, millel on piisavalt tippe (nt **100×100** tipust koosnev) ja mille külje pikus on **10** Blenderi ühikut. Mine muutmisrežiimi Edit (\leftrightarrow Tab), ning lisa tipugruppide paneelis Vertex Groups objekti andmete Object Data all uus tipugrupp. Määra sellesse kõik oma võre tipud (nende kaal võiks näiteks olla **1.0**). Mine tagasi objektirežiimi Object.

Seejärel mine töötletaja omaduste Modifiers all ning lisa tipu kaalude läheduse töötletaja Vertex Weight Proximity. Määra režiimiks objekti kaugus (Object Distance). Vali oma tipugrupp ja soovitatav sihtobjekt (mina kasutasin siin valgustit).

Tõenäoliselt pead sa muutma tipu kaalude läheduse töötletaja poolt loodavate kaalude lineaarset laotust. Selleks muuda väikseima kauguse (Lowest Dist) ja suurima kauguse (Highest Dist) välju nii, et esimene vastab sihtobjekti ja nende tippude, mille puhul sa soovid saada väikseimat kaalu, vahelisele kaugusele ning teine sihtobjekti ja suurima kaaluga tippude vahelisele kaugusele.

Nüüd lisa lainetöötletaja Wave, määra selle sätted oma äranägemise järgi ja pane seda muutma see sama tipugrupp.

Animeeri oma sihtobjekti nii, et see liiguks üle ruudustiku võre. Nagu näed, on lained nähtavad ainult sihtobjekti ümber! Pane tähele, et võid lisada enne lainetöötletajat tipu kaalude muutmise töötletaja Vertex Weight Edit ning kasutada selle spetsiaalkõvera laotust Custom Curve, et tekiksid suuremad/kitsamad "laine mõju nõlvad".

[\[video link\]](#)

[Blenderi fail](#), TEST_1 stseen.

Sihtobjekti geomeetria kauguse kasutamine

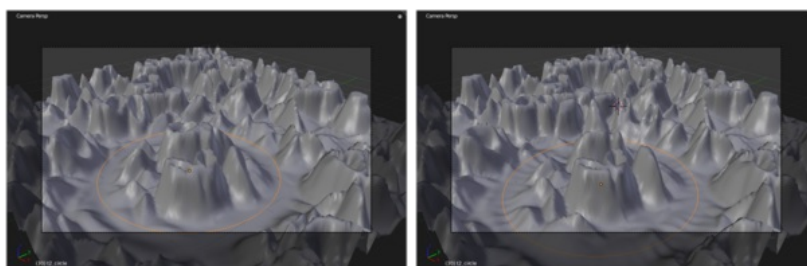
Seda illustreerime nihketöötletajaga Displace.

Lisa **10×10** Blenderi ühiku ja **100×100** tipu suurune ruudustiku võre ning määra see muutmisrežiimis Edit mõnda tipugruppi, mis sisaldaks kõiki selle tippe, nagu ülalpool. Sa võid seda lausa esmalt tükeldamistöötletaja Subsurf abil veelgi detailsemaks muuta.

Nüüd lisa kõverast ring ning aseta see **0.25** Blenderi ühiku kaugusele ruudustiku kohale. Suurenda natuke selle mõõtkava (nt **4.0**).

Lisa ruudustiku võrele tipu kaalude läheduse töötletaja Vertex Weight Proximity ning määra selle režiimiks geomeetria kaugus (Geometry Distance). Lülita sisse servad (Edge, kui kasutaksid ainult tippe (Vertex) ja su kõvera U on madala määratlusega, saaksid sa tulemuseks lainelise mustri nagu pildil (*Lainelised mustrid*)).

Lainelised mustrid.



Kaugus servadest.

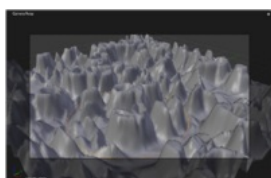
Kaugus tippudest.

Määra väikseima kauguse Lowest Dist väärtuseks **0.2** ja suurima kauguse Highest Dist väärtuseks **2.0**, et väljaarvutatud kaugused sobiksid tavapärasesse kaalude vahemikku.

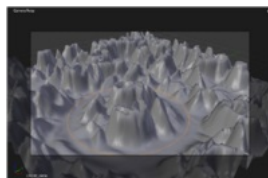
Lisa kolmas nihketöötletaja Displace ning pane see tekstuuri mõjutama vastavalt oma soovile. Nüüd soovime, et kõverast ringile kõige lähemal olevaid tippe ei nihutataks. Kuna nende kaalude väärtus saab olema nullilähedane, määrame nihke keskmise taseme (Midlevel) väärtuseks **0.0**. Paneme selle kasutama meie mõjutatavat tipugruppi ja olemegi valmis! Kaunid mäed kahanevad kõverast ringi läheduses tasapinnaks.

Nagu eelmise näite puhulgi, või enne nihketöötlejat lisada tipu kaalude muutmise töötleja Vertex Weight Edit ning muuta spetsiaalkõvera laotust Custom Curve, et "orud" oleks suuremad/kitsamad...

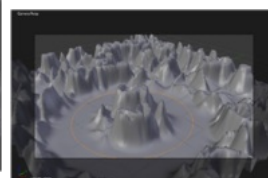
Kõveralaotuse Curve Map variatsioonid.



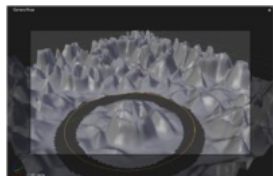
Nõgus laotuskõver.



Laotuskõver puudub (lineaarne).



Kumer laotuskõver.



Tipud, mille arvatud kaal on väiksem kui **0.1**, on tipugrupist eemaldatud.

Samuti võid lisada maskitöötleja Mask ning lülitada sisse tipu kaalude muutmise töötleja Vertex Weight Edit grupist eemaldamise sätte Group Remove, mille eemaldamise läve Rem Threshold väärtuseks on **0.1**, et oma oru põhi üldse kaduma panna.

[\[video link\]](#)

[Blenderi fail](#), TEST_2 stseen.

Tekstuuri ja laotuskõvera kasutamine

Nüüd tekitame mingit sorti imeliku tulnukate laine (jah veel üks lainetöötlejat Wave kasutav näide, kuid see on väga visuaalne ning seetõttu on sellelt lihtne näha tipugrupi efekte).

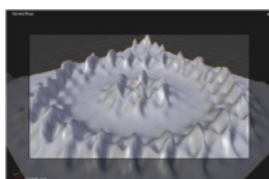
Nagu ennegi, lisa **100×100** ruudustiku võre. Seekord lisa tipugrupp, kuid ära määra sellesse ühtegi tippu – teeme seda dünaamiliselt.

Lisa esimene tipu kaalude segamise töötleja Vertex Weight Mix, määra tipugrupp A Vertex Group A välja vaikimisi A kaalu Default Weight A väärtuseks **0.0** ja vaikimisi B kaalu Default Weight B väärtuseks **1.0**. Jäta segamisrežiimi Mix Mode valikuks tippude asendamine (Replace weights) ja määra segamise komplektiks Mix Set kõik tipud (All vertices). Nõndaviisi mõjutatakse kõiki tippe. Kuna mõjutatavat tipugruppi ei ole, on kõigi tippude väärtus **0.0** ning see asendatakse teise vaikimisi kaaluga (**1.0**). Ning kõik need tipud lisatakse mõjutatavasse tipugruppi.

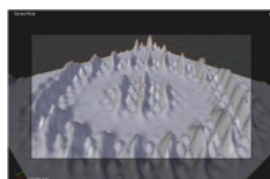
Nüüd vali või loo maskimistekstuur – antud juhul kasutasin mina maagilist vaiketekstuuri Magic. Selle tekstuuri väärtused määravad, kui palju "teine kaal" (**1.0**) asendab "esimest kaalu" (**0.0**) ehk teisisõnu võetakse neid kui kaalu väärtusi!

Seejärel võid valida, milliseid tekstuurikoordinaate ja millist kanalit kasutada. Määra laotuseks lokaalne valik Local ning katseta erinevaid kanaleid.

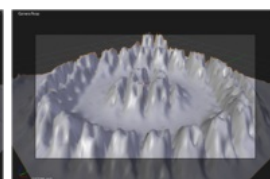
Tekstuurikanali variatsioonid.



Kasutades intensiivsust.



Kasutades punast.



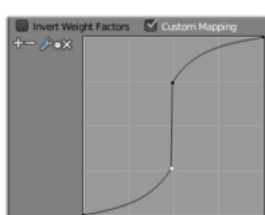
Kasutades küllastust.

Ära unusta lisamast lainetöötlejat Wave ning selles enda tipugruppi valimast!

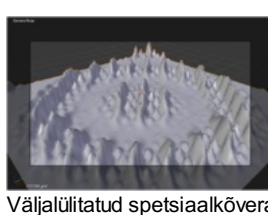
Saad kasutada sellisel moel loodud kaale otseselt, kuid kui soovid laotuskõveraga katsetada, pead lisama kuulsa tipu kaalude muutmise töötleja Vertex Weight Edit ning sisse lülitama spetsiaalkõvera laotuse Custom Curve.

Vaikimisi on see üks-ühele lineaarne laotus ehk siis ei tee mitte midagi! Muuda see millekski pildil (*Spetsiaalne laotuskõver*) oleva samaseks, mis laotab vahemiku [0.0, 0.5] vahemikku [0.0, 0.25] ning vahemiku [0.5, 1.0] vahemikku [0.75, 1.0], luues nõnda peaaegu ainult kaale alla **0.25** ja üle **0.75**: see tekitab lainete sees "seinu".

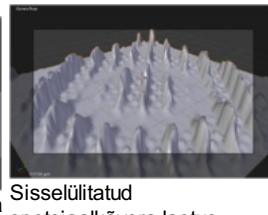
Spetsiaalne laotuskõver.



Is tehtud laotuskõver.



Väljalülitatud spetsiaalkõvera laotus Custom Mapping.



Sisselülitatud spetsiaalkõvera laotus Custom Mapping.

[\[video link\]](#)

[Blenderi fail](#), TEST_4 stseen.

Vaata ka

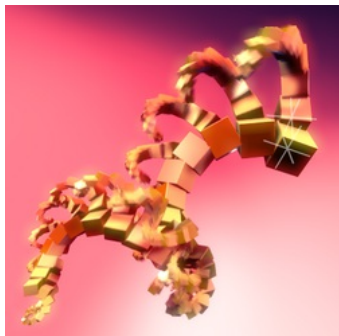
- [arenduslehekülge](#).

Array Modifier - reatöötaja

Mode: Objektirežiim (*Object Mode*)

Panel: *Properties Window* (seadistuste aken) -> Kontekstinupp Modifiers (töötledjad) 

Kirjeldus



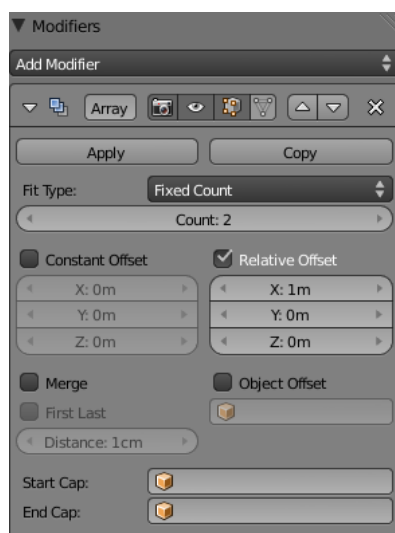
Mitmemõõtmeline rida animeeritud liikumishäguga (*motion blur*).

Array Modifier (reatöötaja) loob baasobjektist koopiate rea (massiivi), mille iga element võib olla eelneva suhtes mitmel viisil ümber paigutatud. Kõrvuti asetsevate koopiate tippe on võimalik ühendada vastavalt määratud ühenduskaugusele. See võimaldab luua sujuvaid jagatud pindu.

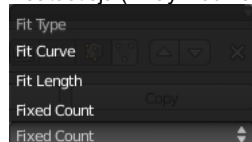
Seda töötajat on praktiline kasutada koos omavahel sobituvate võredegaga, võimaldades kiiresti luua mahukaid stseene. Samuti saab sellega tekitada keerukaid korduvatest elementidest moodustatud kujusid.

Mitu reatöötajat võivad olla ühekorraga aktiivsed – näiteks et moodustada 3-mõõtmelisi konstruktsioone.

Seaded



Reatöötaja (Array Modifier).



Menüü Fit Type (sobitusviis).

Menüü Fit Type (sobitusviis)

Määrab, kuidas määratakse rea pikkus. On kolm valikut, mille igaühel on omad seaded Curve (kõver), Length (pikkus) ja Count (arv):

- Fit Curve (vastavalt kõverale) – loob piisavalt koopial, et mahutada need valikuga Curve (kõver) määratud kõverjoonele.
- Fit Length (vastavalt pikkusele) – loob piisavalt koopial, et mahutada need valikuga Length (pikkus) määratud vahemaa sisse.
- Fixed Count (määratud arv) – loob valikuga Count (arv) määratud arvu koopial.

Curve (kõver)

Kõverjoone objekt, mida kasutatakse Fit Curve (vastavalt kõverale) meetodi puhul.

Length (pikkus)

Vahemaa, mida kasutatakse Fit Length (vastavalt pikkusele) meetodi puhul.

Count (arv)

Koopiate arv, mida kasutatakse Fixed Count (määratud arv) meetodi puhul.

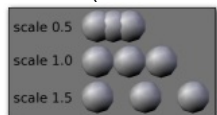
Märkused

- Nii Fit Curve (vastavalt kõverale) kui ka Fit Length (vastavalt pikkusele) kasutavad objekti kohalikku koordinaatsüsteemi. See tähendab, et baasobjekti skaleerimine objektirežiimis Object ei muuda reatöötaja poolt loodud koopiate arvu. Sellisel juhul võib olla vajalik skaleerimine objektile rakendada (Apply Scale).
- Fit Length (vastavalt pikkusele) kasutab kõvera pikkust tema kohalikus koordinaatsüsteemis. See tähendab, et kõvera skaleerimine objektirežiimis Object ei muuda reatöötaja poolt loodud koopiate arvu. Ka sellisel juhul võib olla vajalik skaleerimine kõverale rakendada (Apply Scale).

Constant Offset (konstantne nihe), X, Y, Z

Liidab iga duplitseeritud objekti asukohale täiendava konstantse nihke. X, Y ja Z nihked saab määrata eraldi.

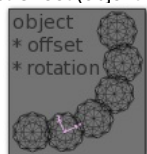
Relative Offset (suhteline nihe), X, Y, Z



Suhtelise nihke (*relative offset*) näide.

Liidab iga duplitseeritud objekti asukohale täiendava nihke, mis igal teljel võrdub duplitseeritud objekti piirdkasti suuruse ja vastavale teljele määratud suhtelise nihke suuruse korrutisega. X, Y ja Z koefitsiente saab eraldi määrata. Vaata pilti (*Suhtelise nihke (relative offset) näide*).

Object Offset (objektiteisendus)



Objektiteisenduse (*object offset*) näide.

Teisendus on võrdne baasobjekti ja määratud objekti distantsiga ning arvestab pööramisega. Vaata pilti (*Objektiteisenduse (object offset) näide*). Hea praktika on kasutada tühiobjekti, mille kese on paigutatud baasobjekti lähedusse. Näiteks tühiobjekti pöörates saab moodustada objektidest ringi või heeliksi.

Merge (ühenda)

Kui valik on sisse lülitatud, siis iga duplitseeritud objekti tipud ühendatakse järgmise koopia tippudega, mis asuvad neile lähemal kui määratud vahemaa (distance).

First Last (esimene viimasega)

Kui see valik **ja** Merge (ühenda) on mõlemad sisse lülitatud, siis esimese objekti tipud ühendatakse viimase koopia tippudega (see võimaldab luua ringikujulisi objekte - vaata pilti (*Esimese ja viimase (First Last) ühendamise näide*)).



Pinnatükelduse probleem, mis tekib, kui esimene ja viimane koopia ei ole ühendatud (First Last (esimene viimasega) on välja lülitatud).

Esimese ja viimase (First Last) ühendamise näide.



Esimese ja viimase koopia ühendamine (First Last (esimene viimasega) on sisse lülitatud) muudab pinnatükelduse pidevaks.

Distance (vahemaa)

Määrab maksimaalse vahemaa, mille puhul tipud ühendatakse (Merge).

Start cap (algusobjekt)

Võre, mida kasutatakse esimese objektina. Üks koopia sellest objektist lisatakse duplitseeritud rea ette, positsiooni -1, st üks "samm" esimesest "normaalsest" duplikaadist ettepoole. Kui Merge on aktiveeritud ja Start cap (algusobjekt) on esimesele koopiale piisavalt lähedal, siis nad ühendatakse.

End cap (lõppobjekt)

Võre, mida kasutatakse viimase objektina. Üks koopia sellest objektist lisatakse duplitseeritud rea lõppu, positsioonile **n+1** võrreldes baasobjektiga, st üks "samm" viimasest "normaalsest" duplikaadist tahapoole. Nagu Start cap (algusobjekt), ühendatakse ka lõppobjekt viimase koopiaga, kui vastavad valikud on sisse lülitatud...

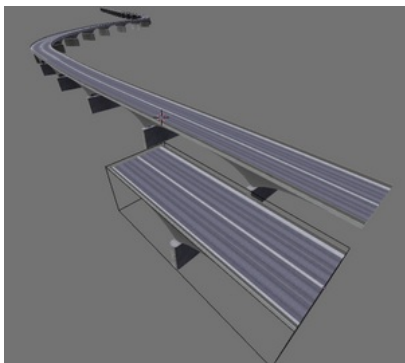
Vihjeid

Teisenduse arvutamine

Teisendus ühest koopiast järgmiseks arvutatakse kolme alamteisenduse summana (Relative (suhteline), Constant (konstantne) ja Object (objektiteisendus)). Neid kõiki saab sõltumatult teistest sisse ja välja lülitada. Näiteks kui määrata suhtelise nihke väärtuseks **(1, 0, 0)** ja konstantse nihke väärtuseks **(0.1, 0, 0)**, võimaldab see luua rea objekte, mis on üksteisest X-teljel eraldatud ühtlaste **0.1BU** (Blenderi ühiku - *Blender Unit*) pikkuste vahedega, sõltumata baasobjekti suurusest.

Näited

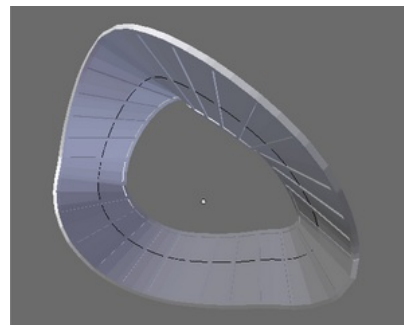
Mehaanilised



Omavahel liituvatest võredest moodustatud sild.

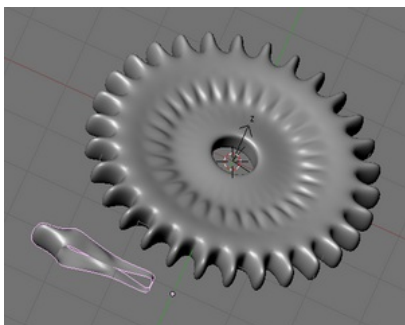
Märkus

Kuna kõveratöötaja (Curve Modifier) ei saa (selle pildi loomise ajal) töötajate pinus asetseda peale reatöötajat (Array Modifier), oli vajalik reatöötaja rakendada (st vajutada nuppu Apply), enne kui sillale kõveratöötaja lisati.



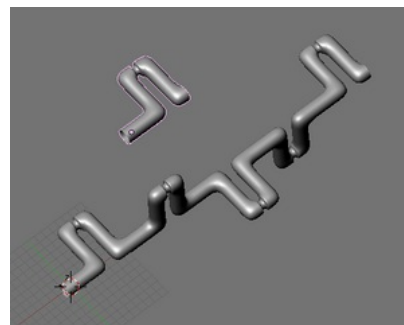
Võidusõidurada.

[.blend-näitefail](#)



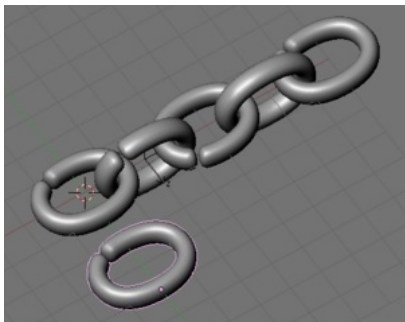
Ühest segmendist loodud hammasratas.

[.blend-näitefail](#)



Väntvõll.

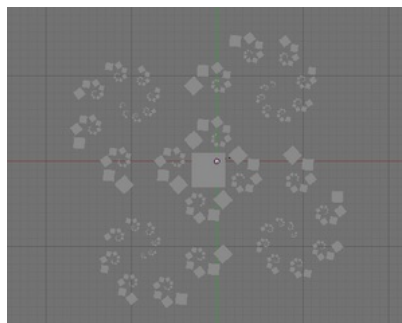
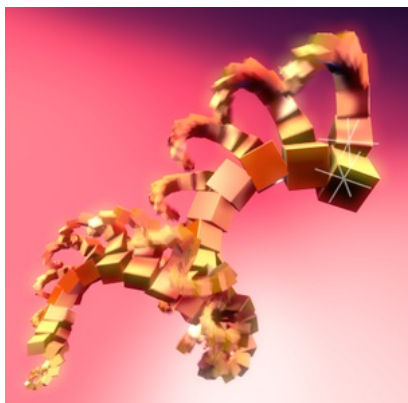
[.blend-näitefail](#)



Ühest lülisest duplitseeritud kett.

[.blend-näitefail](#)

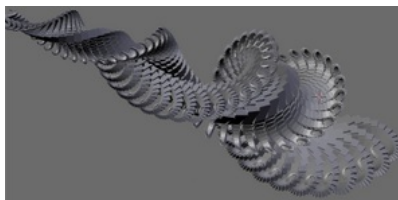
Fraktaalised



Fraktaalismane pilt, mis on genereeritud kuubile rakendatud mitme reatöötaja abil.

[.blend-näitefail](#)

Mitmemõõtmeline rida liikumishäga.

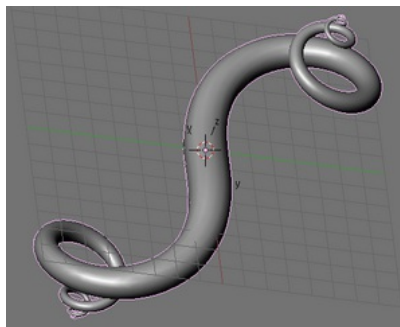


Fraktaalne sõnajalg, mis on genereeritud kuubile rakendatud 2 reatõõtleja ja 1 peegelduse abil.

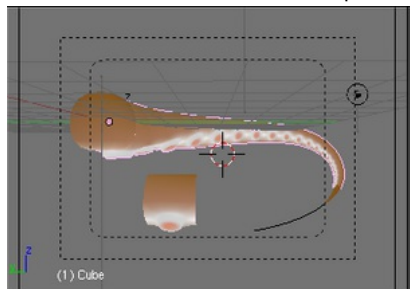
Orgaanilised



Tükeldatud pinnaga kuupide rida, mis on moodustatud 4 kuubist 1 objektiteisenduse ja suure tippude ühendamise (*Merge*) distantse abil, loob sileda katkematu pinna.



Kaksikspiraal, mis on moodustatud kuubile rakendatud 2 reatõõtleja ja pinnatükelduse abil. Nagu ka eelmises näites, on nahaefekti loomiseks tippude ühendamise (*Merge*) minimaalne distantse suur. [.blend-näitefail](#)



Kombits, mis on moodustatud reatõõtleja (Array Modifier) ja sellele järgneva kõveratõõtleja (Curve Modifier) abil. Esiplaanil olev segment on kombitsa baasvõre; otstes paiknevad kaks spetsiaalselt loodud objekti, mida deformeerib seesama kõveratõõtleja, mis ülejäänud kombitsat. [.blend-näitefail](#)

Õppetükid

Õppetükid

- [Neal Hirsig'i ekraanivideo reatõõtlejast](#)
- [Kaksikheeliksi loomine töötlejate abil](#)

'Kaksikheeliksi' õppetükk selgitab reatõõtleja (Array Modifier) kasutamist. Videos kasutatakse vana Blenderi versiooni (2.44), kuid peale kiirklahvide kehtib kõik ka uues versioonis.

Bevel Modifier - kantimistöötleja

Mode: Objektirežiim (*Object Mode*)

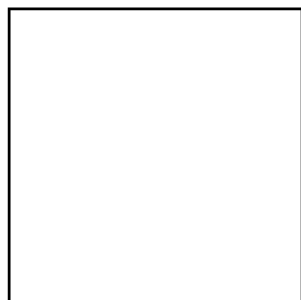
Panel: *Properties Window* (seadistuste aken) -> Kontekstinupp Modifiers (töötledjad) 

Kirjeldus

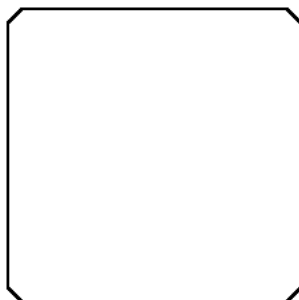
Kantimistöötleja (*Bevel Modifier*) võimaldab võre servi kantida (faasida). Seda, milliseid võre osasid ja kuidas kantida, on võimalik eraldi määrata.

Mis on kantimine (*bevel*)?

Definitisioon (inglisekeelses) Wikipedias: [bevel](#).



Kantimata ruut.



Kanditud ruut.

Pildil (*Kantimata ruut*) on teravate nurkadega ruut, st nurgad servade vahel on 90° (täisnurgad). Pildil (*Kanditud ruut*) on ruudu nurgad kantitud (faasitud).

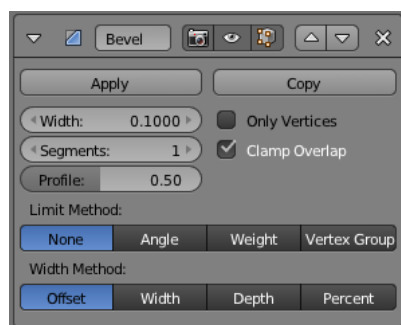


Vaikimisi kantimine.

Ehkki ülaltoodud pildidel on kujutatud 2D-ruudud, võimaldab Blenderi *Bevel Modifier* kasutada kõiki 2D- ja 3D-võresid, mitte ainult ruute ja kuupe...

Pildil (*Vaikimisi kantimine*) on näidatud Blenderi 3D-kuup, mida on kantitud kantimistöötleja vaikimisi seadetega.

Valikud



Bevel Modifier (kantimistöötleja) paneel.

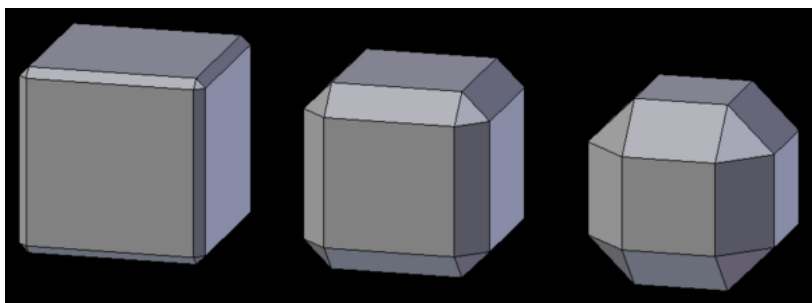
Kantimistöötleja paneel on selge paigutusega ja peaks olema intuiitselt mõistetav. Järgnevalt kirjeldatakse võimalikke valikuid:

Width - Laius

Numbriväli *Width* (laius) määrab baasvõrele rakendatava kandi laiuse (suuruse). Võimalikud väärtused on **0.0** (kantimist ei toimu) kuni **1.0** Blenderi ühikut. See väärtus määrab kahe uue serva suhtelise kauguse algsest (kantitavast) servast piki vastavaid külgi.

Märkus

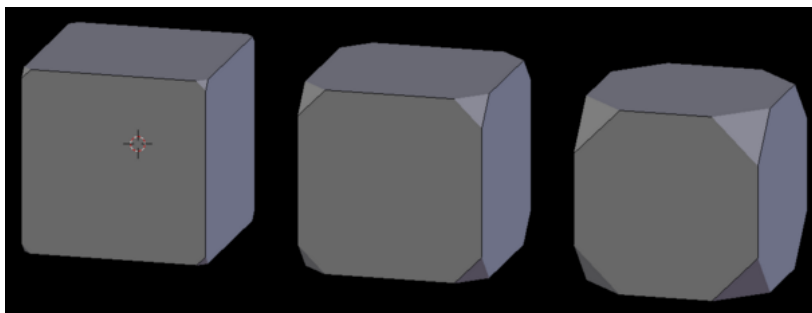
Kui kasutate meeter- või tollimõõdustikku, on laiusel ka ühik. Näiteks kui 1 Blenderi ühik on 1 m, on kandi laiuse kasulikud väärtused 0 cm kuni 100 cm. Kui näib, et laiuse vähendamine ei mõjuta kandi suurust, kontrolli, et ühikuks ei oleks sattunud m (cm asemel).



Kolm kuupi **0,1**, **0,3** ja **0,5** laiuse kandiga.

Ainult tipud

Nupp Only Vertices (ainult tipud) muudab viisi, kuidas kantimist võrele rakendatakse. Kui nupp on aktiivne, kantitakse ainult piirkondi tippude ümber, jättes servad kantimata.



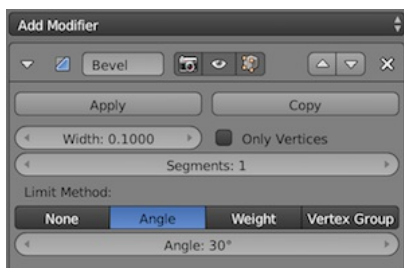
Kolm kuupi **0,1**, **0,3** ja **0,5** laiuse kandiga, kui valik Only Vertices (ainult tipud) on sisse lülitatud.

Limit Method - Piiramisviis

Sellega määratakse, millistele baasvõre osadele ja kuidas kantimistöötlejat rakendatakse. Esimesed kolm nuppu (millest korraga saab aktiivne olla ainult üks) määravad kasutatava algoritmi ja aktiveerivad mõningad lisaseadistused.

None (puudub)

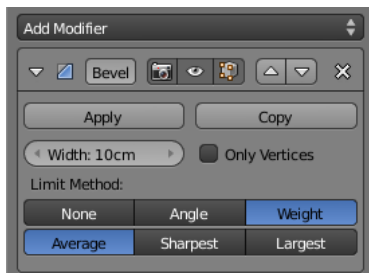
Selle valikuga rakendatakse töötleja tervele võrele ilma ühtegi serva ega tippu välja jätmata.



Bevel Modifier sisse lülitatud nurgapiiranguga (Angle).

Angle (nurk)

Selle valikuga kantitakse ainult need servad, kus küljed moodustavad terava nurga. Kui see nupp on aktiveeritud, kuvatakse numbrisestusväli Angle (nurk), kus saab määrata minimaalse nurga, millele kantimist rakendatakse (tegelikult on tegemist täiendnurgaga, st $180^\circ - (\text{nurk külgede vahel})$). Kui nurk kokkupuutuvate külgede vahel on väiksem kui välja väärtus, siis serva ei kantita. Samaselt: kui nurk servade vahel on lüvendist väiksem, siis tippu ei kantita.



Bevel Modifier sisse lülitatud piiranguga Weight (kaal).

Weight (kaal)

...

Õppetükid mujal

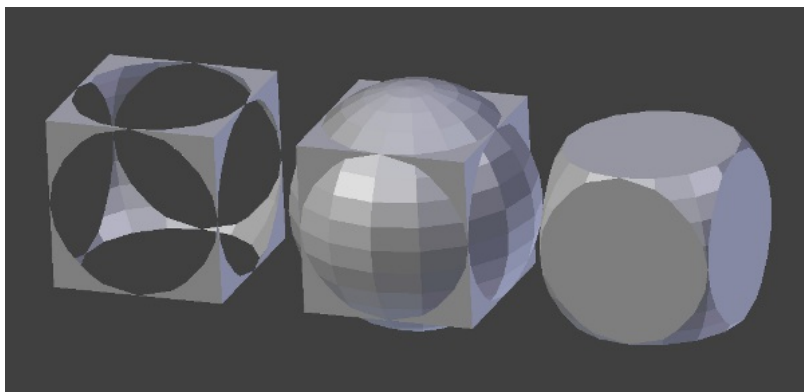
- [Neal Hirsig'i ekraanivideo kantimistöötlejast](#) link ei tööta!

Boolean Modifier - kahendtöötaja

Mode: Kõik režiimid

Panel: Modifiers (Töötledjad)

Kirjeldus



Kuubi ja UV-sfääri erinevus, ühend ja ühisosa, kui töötaja on lisatud kuubile

Boolean Modifier (kahendtöötaja) loob kahest võrest (Mesh) uue kombineeritud võre, kasutades ühte kolmest kahendoperatsioonist: (Difference (erinevus) - eitus, Union (ühend) - disjunktsioon ja Intersect (ühisosa) - konjunktsioon).

Valikud



Boolean Modifier
(Kahendtöötaja)

Operation (meetod)

Difference (erinevus)

Sihtvõre lahutatakse muudetavast võrest.

Union (ühend)

Muudetav ja sihtvõre liidetakse kokku.

Intersect (ühisosa)

Tulemuseks on muudetava ja sihtvõre kattuv osa.

Object (objekt)

Sihtobjekti nimi. See peab olema võre.

Vaata ka

- [Doc:Manual/Modifiers/Mesh/Booleans](#) (Blender 2.4 võre kahendtöötaja kirjeldus koos näidetega (ingliseelne))

Build Modifier - ehitaja

Mode: Object Mode (Objektirežiim)

Panel: Modifiers (Töötlejad)

Kirjeldus

Build Modifier (ehitaja) loob animatsiooni, mille jooksul võre küljed ilmuvad ühekaupa. Kui võre materjal ei ole tavaline materjal, vaid halo, ilmuvad ühekaupa tipud, mitte küljed.

Vaikimisi ilmuvad küljed (tipud) selles järjekorras, nagu nad on programmi mälus (vaikimisi on see loomise järjekord). Seda järjekorda saab muuta muutmisrežiimis (Edit Mode), kasutades käsku Sort Faces (külgede sorteerimine) (CtrlAltF).

Valikud



Build Modifier (ehitaja)

Start (algus)

Kaader, milles alustatakse ehitamist.

Length (pikkus)

Kaadrite arv, mille jooksul objekt üles ehitatakse.

Randomize (muuda juhuslikuks)

Muudab külgede ilmumise järjekorra juhuslikuks.

Seed (seeme)

Juhuarvude generaatori seeme. Muuda seda, et tekiks uus "juhuslik" ilmumisjärjekord – sama seemne-objekti paari kasutamisel on järjekord alati ühesugune.

Decimate Modifier - lihtsustaja

Mode: Object Mode (Objektirežiim)

Panel: Modifiers (Töötledjad)

Kirjeldus

Lihtsustaja (Decimate Modifier) võimaldab võre tippude/külgede arvu võimalikult väheste moonutustega vähendada. Seda ei ole mõtet rakendada hoolikalt ja kasinalt modelleeritud võrele, mille kõik tipud ja küljed on õige kuju määramiseks vajalikud. Kui aga võre on mitmeetapilise modelleerimise tulemus, kus on kasutatud proportsionaalset muutmist, üksteisele järgnevaid täiendusi ja võib-olla ka teisendamist tükeldatud (*SubSurf*) ja tükeldamata pinnaga võrede vahel, tekib tihti võre, mille paljud tipud ei ole kuju määramiseks vajalikud.

Lihtsustaja (Decimate Modifier) võimaldab lihtsalt ja ilma algset võre muutmata vähendada võre külgede arvu. See on näide töötlejate süsteemi eelistest - operatsioone, mis varem muutsid jäävalt algset võret, saab nüüd töötlejate abil rakendada interaktiivselt ja ohutult.

Erinevalt enamikest teistest töötlejatest, ei võimalda lihtsustaja muudatusi muutmisrežiimis (Edit Mode) visuaalselt jälgida.

Decimate mõjub ainult kolmnurkadele, seega muudetakse kõik nelinurksed küljed enne lihtsustamist vaikimisi kaheks kolmnurgaks.

Valikud



Decimate Modifier
(lihtsustaja)

Ratio (suhe)

Määrab, kui palju külgi pärast lihtsustamist objektile jääb. Võimalikud väärtused **0,0** (0%, kõik küljed eemaldatakse) kuni **1,0** (100%, säilitab esialgse võre, kõik nelinurgad kolmnurkadeks jagatuna). Suhet **1,0**-st kuni **0,0**-ni vähendades lihtsustub võre tundmatuseeni.

Face Count (külgede arv)

See mittemuudetav väli näitab, kui palju külgi muutja Decimate rakendamisel võrele alles on jäänud.

Näited

Lihtne tasand

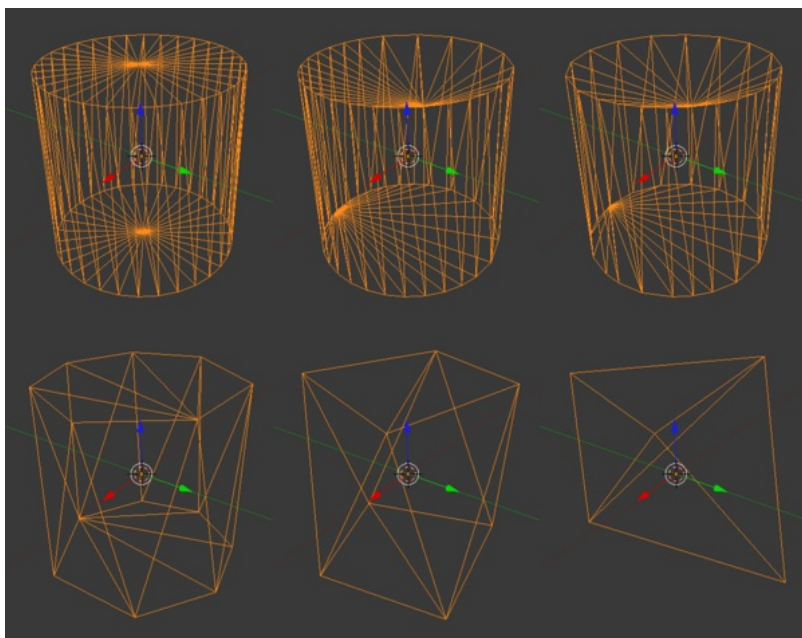
Lihtne näide on tasand ja 4×4 deformeerimata Grid (ruudustik). Mõlemad näevad täpselt ühesugused välja, kuid tasandil on üks külg ja neli tippu, samas kui ruudustikul on 9 külge ja 16 tippu. Seega on enamik ruudustiku tippudest ja külgedest mittevajalikud. Lihtsustaja võimaldab need mittevajalikud küljed eemaldada.

Lihtsustatud silinder

Võtame näite, kuidas lihtsustada silindrit, mis Blenderis koosneb vaikimisi **32** segmendist. Sellisel silindril on **96** külge. Kui rakendada lihtsustajat, siis esialgu külgede arv tõuseb! See juhtub sellepärast, et kõik nelinurgad (*quads*) muudetakse kolmnurkadeks (*tris*), millega alati suureneb külgede arv. Igast nelinurgast tekib kaks kolmnurka.

Lihtsustaja peamine ülesanne on vähendada võre andmemahtu, eemaldades külgesid ja tippe, säilitades samal ajal objekti esialgse väljanägemise.

Igal järgneval pildil on suhte (ratio) väärtust järjest vähendatud - **100%**-st kuni **5%**-ni (0,05). NB! Külgede arv on suhte **0,6 (60%)** puhul kahanenud **128**-lt **88**-le, samas kui silinder on ikka väga samane esialgsele. Seega on kõrvaldatud **40** mittevajalikku külge.



1.0 (100%). külgi: 128; 0,8 (80%). külgi: 102; 0,6 (60%). külgi: 88
0,2 (20%). külgi: 24; 0,1 (10%). külgi: 12; 0,05 (5%). külgi: 6

Nagu näha, hakkab silinder meenutama kuupi, kui suhe läheneb väärtusele **0,1**. Kui suhe jõuab väärtuseni **0,05**, ei meenuta ta isegi enam kuupi!

Kui oled jõudnud sobiva külgede arvu ja väljanägemiseni, võid töötleja rakendada nupuga Apply. Kui tahad täiendavaks võre andmemahu vähendamiseks ühendada võimalikult palju kolmnurki tagasi nelinurkadeks, mine muutmisrežiimi Edit Mode, vali kõik tipud (A) ja vajuta klahvikombinatsiooni AltJ.

EdgeSplit Modifier - servalõhestaja

Mode: Kõik režiimid

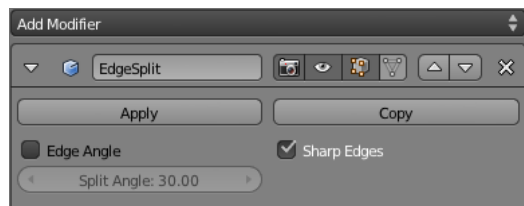
Panel: Modifiers (Töötlejad) (Muutmiskontekst (Editing context), F9)

Kirjeldus

Servalõhestaja (EdgeSplit Modifier) katkestab määratud servadel võre pinna pidevuse. Servad, mida lõhestada, saab valida serva nurga (st nurga serva moodustavate külgede vahel) järgi ja/või märkides ära teravad (*sharp*) servad.

Serva lõhestamine mõjutab servadel tipunormaalide arvutamist, andes vastavale nurgale terava väljanägemise. Seega saab seda töötlejat kasutada sama efekti saavutamiseks, kui nuppu Autosmooth (automaatne silumine), muutes teravaks need servad, mida moodustavate külgede vaheline nurk on suurem teatud väärtusest. Servalõhestaja abil saab silumisprotsessi käsitsi täpselt juhtida, määraates, millised servad peavad näima siledad ja millised teravad (vaata ptk [Võre silumine](#) teisi viise, kuidas sama efekti saavutada). Kui vaja, saab mõlemad lõhestusrežiimid korraga sisse lülitada. Servalõhestaja väljund on kättesaadav eksportskriptidele, mis muudab selle praktiliseks ka mängude sisu modelleerimisel.

Valikud



Servalõhestaja (EdgeSplit Modifier)

From Edge Angle (serva nurgast lähtuvalt)

Kui see on sisse lülitatud, siis servad, mille nurk on suurem kui lõhestusnurga (Split Angle) väärtus, lõhestatakse.

- Serva nurk on nurk kahe külje vahel, mis selle serva moodustavad.
- Kui serv kuulub rohkem kui kahele küljele, on ta alati lõhestatud.
- Kui serv kuulub ainult ühele küljele, ei ole ta kunagi lõhestatud.

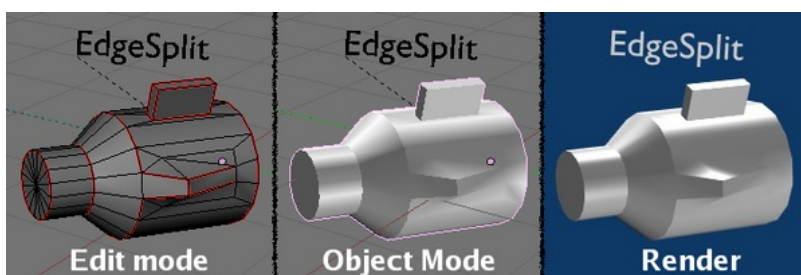
Split Angle (lõhestusnurk)

Minimaalne servanurk, mille puhul serv lõhestatakse, kui From Edge Angle (alates servanurgast) nupp on sisse lülitatud. Võimalikud väärtused on alates **0°**-st (kõik servad lõhestatakse) kuni **180°**-ni (ühtegi serva ei lõhestata).

From Marked As Sharp (märkest **terav**)

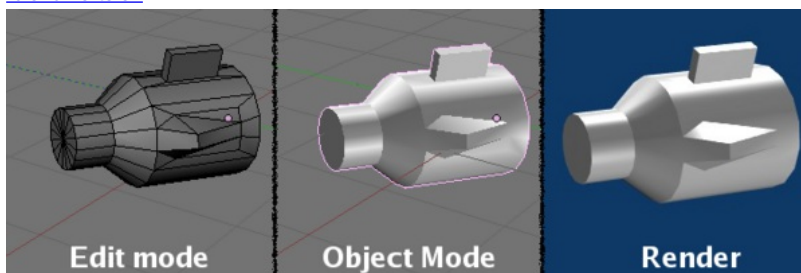
Kui see nupp on sisse lülitatud, lõhestatakse teravaks märgitud servad. Servade teravust saab märkida menüüvaliku Edge Specials » Mark Sharp (märgi teravaks) abil (CtrlE muutmisrežiimis (Edit mode)).

Näited



Servalõhestaja väljund, kui From Marked As Sharp (märgitud teravaks) on sisse lülitatud.

[.blend-näitefail](#)



Servalõhestaja väljund, kui From Edge Angle (alates servanurgast) on sisse lülitatud.

[.blend-näitefail](#)

Mask Modifier - maskitõtleja

Mode: Kõik režiimid

Panel: Modifiers (Tõõtlejad)

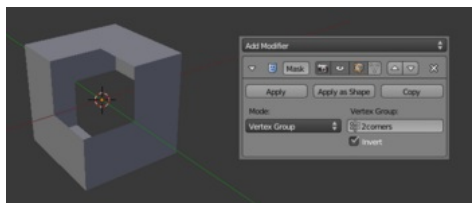
Kirjeldus

Maskitõõtleja (Mask Modifier) võimaldab varjata (maskeerida) mõned võreobjekti osad, justkui puuduksid nad võre koosseisust.

Valikud

Mode (režiim)

Siit menüüst saab valida ühe kahest võre osade peitmise meetodist, mida maskitõõtleja võimaldab:



Vertex Group (tipugrupp)

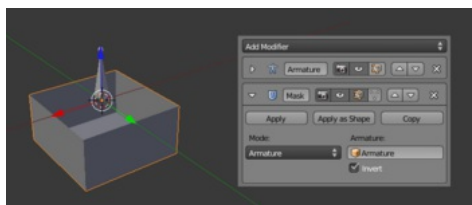
Vertex Group (tipugrupp)

Kui tipugrupp on valitud, mõjutab maskitõõtleja määratud tipugruppi kuuluvate võre osade nähtavust.

Kui soovitud tipugrupi nimi on sisestatud, muudab maskitõõtleja võre nähtavust nii, et vastavasse gruppi kuuluvad tipud on maskeeritud (mis üldjuhul tähendab, et nad on nähtavad) ja need tipud, mis vastavasse gruppi ei kuulu, on nähtamatud.

Kõik tipu kaalude seadmise meetodid töötavad ka maskitõõtlejaga (Mask modifier), kuid tippude kaalude täpseid arvulisi väärtusi ignoreeritakse. Maskitõõtleja arvestab ainult seda, kas vastav tipp kuulub gruppi või mitte, kaal ei oma tähtsust.

Seega kui tipugrupi kaal on näiteks **0.5**, ei tekita see osaliselt maskeeritud võret. Tipu vastavasse gruppi kuulumine on maskitõõtlejale piisav, isegi kui tipu kaal on **0.0**.



Armature (skelett)

Armature (skelett)

Kasulik poseerimisrežiimis (Pose Mode) või skeleti muutmisel. Sellele väljale tuleb sisestada skeletiobjekti nimi.

Töötades luudega poseerimisrežiimis, maskeeritakse need tipugrupid, mis pole seotud aktiivse luuga. Vajutades nuppu Inverse (teistpidi) võib näha, kuidas luu mõjutab võret mujal, mööda luude ahelat edasi.

Inverse (teistpidi)

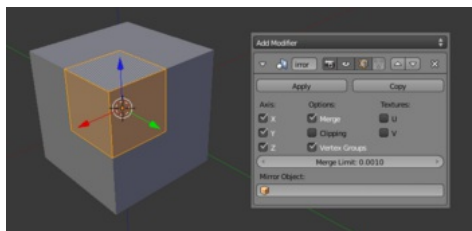
Tavaliselt muudab maskitõõtleja oma mõju all olevad võre osad nähtavaks ja peidab ülejäänud osad. Nupp Inverse (pööra) muudab töötleja käitumise vastupidiseks, st need võra osad, mis olid algselt peidetud, muutuvad nähtavaks ja vastupidi.

Mirror Modifier - peegeldaja

Mode: Kõik režiimid

Panel: Modifiers (Töötledjad)

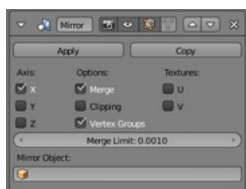
Kirjeldus



Kuubi nurk peegeldatuna piki kolme telge, moodustamaks... ee... noh, kuupi.

Peegeldaja (Mirror Modifier) peegeldab võret automaatselt piki objekti keskpunkti läbivaid **lokaalseid** X-, Y- ja/või Z-telgi (kaks ülejäänud telge moodustavad peegeldustasapinna). Peegelduse keskpunktina saab kasutada ka mõnda teist objekti, sel juhul kasutatakse peegeldustelgedena selle objekti lokaalseid koordinaattelgi. Peegeldaja võimaldab ühendada baasobjekti ja peegelduse tipud, mis asuvad lähemal kui määratud minimaalne kaugus (*tolerance*). Baasobjekti tippudele nihutamisele saab seada piirangu, mis ei võimalda neid liigutada läbi peegelduspinna. Lisaks sellele võimaldab see töötleja peegeldada ka tipugruppe ja UV-koordinaate.

Valikud



Mirror Modifier
(peegeldaja)

Axis (telg)

Koordinaattelg (X, Y või Z), mida mööda peegeldatakse (st telg, mis on rist peegeldustasapinnaga). Selgituseks selle kohta, kuidas valitud telg mõjutab peegelduse suunda: kui näiteks peegeldusteljena kasutada X-telge, siis algse võre tippude positiivsetest X-koordinaadi väärtustest saavad peegeldatud võrel negatiivsed X-koordinaadi väärtused. Korraga võib valida rohkem kui ühe telje, sel juhul tekib ka rohkem peegeldusi. Iga peegeldustelg lisab töötleja tulemusel tekkivale võrele täiendava sümmeetriasapinna, st üks telg tekitab 2 objekti (algne ja üks peegeldus), kaks telge 4 objekti (algne ja 3 peegeldust) ja kolm telge 8 objekti (algne ja 7 peegeldust).

Valikud

Merge (ühenda)

Ühendab punktid peegeldustasapinnal (-pindadel). Vaata seadistust Merge Limit (ühendamise kaugus) allpool.

Clipping (piiramine)

Tõkestab tippude liikumise läbi peegelduspindade. Märkus: see kehtib ainult muutmisrežiimis (st kui muuta objekti teisendust, nihet, skaalat jne objektirežiimis, võivad tipud vabalt peegeldustasapindu ületada.)

Kui Clipping (piiramine) on sisse lülitatud, aga tipud on kaugemal kui Merge Limit (ühendamise kaugus), siis tippe ei liideta. Niipea kui tipud jõuavad seadega Merge Limit (ühendamise kaugus) määratud kaugusele, ühendatakse nad peegeldustasapinnal kokku ja neid saab edasi liigutada ainult mööda peegelduspinda. Kui mitu erinevat tippu, mis on peegelduspinnast erinevatel kaugustel, on korraga valitud, rakendatakse piiramist neile ühekaupa. Kui oled piiratud tippude asukoha korra kinnitanud LMB ja tahad seejärel tippe peegeldustasapinnalt eemale nihutada, tuleb piirang enne lõpetada, lülitades valiku Clipping (piiramine) paneelil välja.

Vertex Groups (tipugrupid)

Kui see on sisse lülitatud, püüab peegeldaja (Mirror Modifier) peegeldada ka võre tipugruppe. See on väga kasulik meetod, kuid sel on teatud spetsiifilised eeldused.

- Kõigepealt: tipugruppide, mida sa tahad peegeldada, nimed peavad järgima tavapärast vasak/parem mustrit (st lõppema märgistega nagu ".R", ".right", ".L").
- Järgmiseks: "peegeldatud" grupid peavad juba olemas olema (st samade nimede ja "teise poole" lõppmärgistega) ja olema täiesti tühjad (mitte sisaldama ühtegi tippu), vastasel korral tipugruppide peegeldamine ei tööta.

Tavalisel juhul jäävad tippude peegeldatud koopiad samasse gruppi nagu algseid tipud. Kui aga tipugruppide valik on aktiveeritud, kuuluvad kõigile ülaltoodud tingimustele vastavad grupid algsele objektile ja sisaldavad selle tippe: peegeldusel paigutatakse vastavad tipud "peegeldatud" gruppi. See on väga praktiline skeleti puhul. Näiteks modelleri ainult pool objektist, taagelda (*rig*) see hoolikalt poole skeletiga ja lihtsalt lase töötlejal (Mirror Modifier) genereerida puuduv pool. Veendu, et kõik skeletitöötledjad (Armature Modifier) oleks töötlejate pinus tagapool peegeldajat. Viimane repliik piki mitut telge peegeldamise kohta: sel juhul saavad "otsesed", "esimese taseme" koopiad endale peegeldatud grupid, koopiaste koopiad ("teine tase") jälle algseid grupid jne.

Textures (tekstuurid)

Valikud U ja V võimaldavad peegeldada vastavalt võre U- ja V-tekstuurikoordinaate. Väärtused "peegeldatakse" üle **0.5** - st kui sul on tipp UV-koordinaatidega (**0.3**, **0.85**) ja mõlemad nupud on sisse lülitatud, on selle koopia UV-koordinaadid (**0.7**, **0.15**).

Merge Limit (ühendamise kaugus)

Suurim vahemaa tipu ja peegelduspinna vahel, mille korral tipp veel oma peegeldusega ühendatakse. Ühendatavad tipud liiguvad ühte punkti peegelduspinnal, võimaldades algse võre peegeldusega sujuvalt ühendada.

Mirror Object (peegeldusobjekt)

peegelduse juhtimiseks kasutatava teise objekti (tavaliselt tühiobjekti) nimi. Selle objekti keskpunkt ja lokaalsed koordinaatteljed määravad peegelduspinnad. Selle asukohta ja pööret on mõistagi võimalik animeerida (IPO kõveratega või muul viisil), et tekitada animeeritud peegeldusefekti.

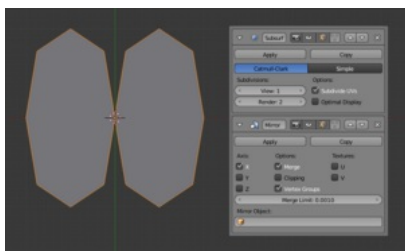
Vihjed

Paljude modelleerimisülesannete jaoks on vaja luua sümmeetrilisi objekte. Varem polnud aga kiiret meetodit, modelleerimaks korraka objekti mõlemaid pooli, muidu kui kasutades kavalate blenderikasutajate poolt aastate jooksul avastatud kaudseid meetodeid. Tavaline tehnika oli modelleerida pool objektist ja vajutada siis AltD, et luua seostatud koopia. Seda sai peegeldada piki ühte telge, luues algse objekti täieliku peegelduse, mis reaalses järgis kõiki lähteobjekti muutusi.

Mirror Modifier (peegeldaja) võimaldab seda teha teisel, lihtsamal viisil. Kui modelleerimine on lõpetatud, võid kas vajutada Apply (rakenda), et tekitada terviklik versioon oma võrest, või jätta töötleja paika hilisemaks muutmiseks.

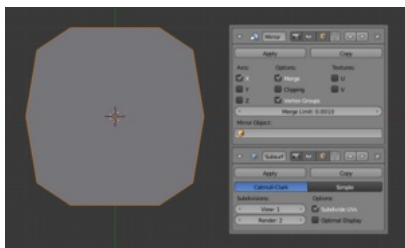
Peegeldaja (*Mirror modifier*) kasutamine koos pinnatükeldajaga (*Subdivision Surface Modifier*)

Kui kasutada peegeldajat pinnatükeldajaga, on nende töötlejate järjekord pinus oluline.



Pinnatükeldaja (Subsurf Modifier) enne peegeldajat (Mirror Modifier)

Sellel pildil on Subsurf asetatud ettepoole töötlejast Mirror. Nagu pildilt näha, tekib objekti peegeldustasapinnal vagu, mis võib objekti ka pooleks jagada.



Peegeldaja (Mirror Modifier) enne pinnatükeldaja töötlejat (Subsurf Modifier)

Sellel pildil on peegeldaja asetatud pinnatükeldajast ettepoole. Selle järjekorra puhul paikneb aluspinna keskjoon peegelduspinnal, mis enamikel juhtudel ongi soovitud tulemus.

Joondamine peegeldamiseks

Et rakendada Mirror Modifier, nihutatakse objekti keskpunkt enamasti servale või küljele, millest saab peegeldustelge. Seda silma järgi teha võib osutuda küllaltki keerukaks. Hea meetod täpse asukoha saavutamiseks on kõigepealt määrata serv, üle mille sa tahad peegeldada. Valida kaks tippu sellel serval. Siis vajutada ⇧ ShiftS ja selle järel *Cursor to Selection* (kursor valitule). See nihutab 3D-kursori täpselt valitud serva keskpunkti. Lõpuks, muutmiskontekstis (Editing Context - F9) valida Mesh (võre) paneelil Center Cursor (tsentreeri kursorile) — see nihutab objekti keskpunkti 3D-kursori asukohta. Nüüd on peegeldamine täpne.

Muidugi võib kasutada ka lihtsalt teist tühiobjekti peegelduskeskpunktina.

Multiresolution Modifier - parralleelvõre

Mode: Objektirežiim (*Object Mode*)

Panel: *Properties Window* (seadistuste aken) -> Kontekstinupp Modifiers (töötledjad) 

Parralleelvõre (Multiresolution Modifier) võimaldab kasutada sama objekti jaoks erineva tihedusega võresid 3D-akna, voolimisrežiimi ja renderdamise jaoks.

Valikud

Catmull-Clark / Simple

Määrab külgede ja servade tükeldamise viisi. Simple (lihtne) säilitab olemasoleva kuju ja jagab ainult servad osalõikudeks.

Catmull-Clark moodustab esialgsest pinnast väiksema silendatud pinna.

Preview (eelvaade)

Määrab pinnatükelduse tasemete arvu 3D-aknas.

Sculpt (voolimine)

Määrab pinnatükelduse tasemete arvu voolimisrežiimis (Sculpt Mode).

Render (renderdus)

Määrab pinnatükelduse tasemete arvu renderdamisel.

Subdivide (tükelda)

Lisab võrele uue (tihedama) tükeldustaseme.

Delete Higher (kustuta tihedamad)

Kustutab kõik tükeldustasemed, mis on suuremad kui parajasti kehtiv.

Reshape (kujunda ümber)

Kopeerib tippude koordinaadid teiselt võrele. Et seda meetodit kasutada, vali teine kattuva topoloogia ja tippude koordinaatidega võre, siis vali ⇧ Shift abil töödeldav võre ja vajuta nuppu Reshape (kujunda ümber). Võre võtab teise võre kuju.

Apply Base (rakenda baasile)

Muudab algse võre kuju samaseks tükeldatud võrele.

Optimal Display (optimaalne esitus)

Ei joonista tükeldamise käigus lisatud servi.

Save External (salvesta väline)

Salvestab tippude ümberpaigutused välisesse .btx-faili.

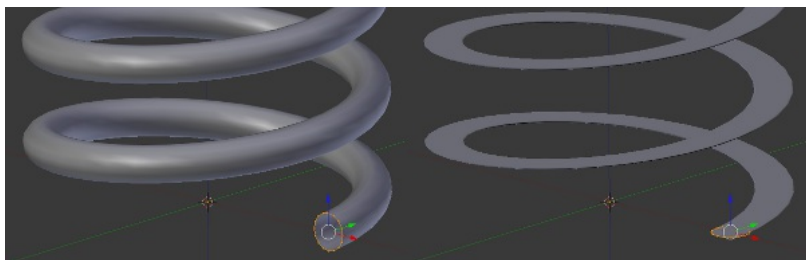
Screw Modifier - kruvitöötleja

Mode: Kõik režiimid

Panel: Modifiers (Töötlejad)

Kirjeldus

Kruvitöötleja (Screw Modifier) samaneb kruvitööriistale (Screw) tööriistakastis (Tool Shelf): see loob algsest objektist (võre või kõver) heeliksilaadse keha.



Profilobjekti õige joondamine on oluline.

Profilobjekt peab olema kahemõõtmeline ja paralleelne (mitte ristuv) kruvi teljega.

Valikud



Screw Modifier -
kruvitöötleja

Axis (telg)

Koordinaattelg, piki mida heeliks ehitatakse.

Screw (kruvi)

Ühe heeliksi iteratsiooni kõrgus (pikkus).

AxisOb (teljeobjekt)

Objekti, mis määrab telje suuna, nimi.

Object Screw (kruvi objekt)

Loe Screw (kruvi) parameetri väärtus teljeobjektilt (Axis Object).

Angle (nurk)

Heeliksi ühe pöörde pöördenurk kraadides.

Steps (samme)

Heeliksi ühe pöörde sammude arv (3D-aknas esitamisel.)

Render Steps

Sama mis eelmine renderdamisel. Suurenda seda väärtust sujuvama heeliksi saamiseks.

Calc Order (arvuta järjekord)

Et vältida probleeme normaalidega, arvutatakse külgede järjekord ümber. See on vajalik ainult võrede, mitte kõverate puhul.

Flip (pööra)

Vahetab normaalide suuna.

Iterations (kordusi)

Iteratsioonide arv kogu heeliksis.

Solidify Modifier - paksuse lisaja

Mode: Kõik režiimid

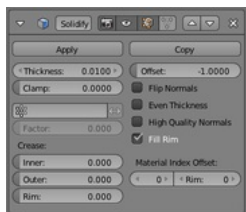
Panel: Modifiers (Töötledjad)

Kirjeldus

Paksuse lisaja (Solidify Modifier) teeb suvalise võre pinna ruumiliseks kehaks.

See töötleja oli algsest kirjutatud Campbell Bartoni (alias Ideasman) poolt iseseisva skriptina ja [lisatud Blenderi versioonile 2.44](#). Blender 2.5 jaoks muudeti ta töötlejaks.

Valikud



Solidify Modifier -
paksuse lisaja

Thickness (paksus)

Pinnast moodustatud keha paksus.

Offset (nihe)

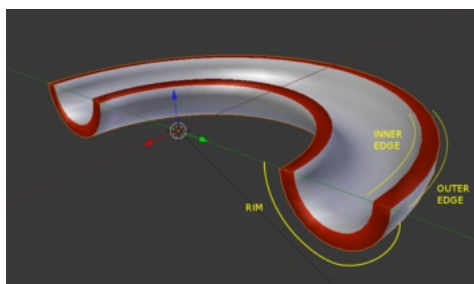
Väärtus **-1** ja **1** vahel määrab, kui suurel määral on loodud keha esialgsest pinnast sees- või väljaspool. Kui Offset (nihke) väärtus on 0, paikneb uue keha keskpunkt algse võre pinna kohal.

Vertex Group (tipugrupp)

Rakenda töötlejat ainult selle grupi tippudele.

Invert (pööra)

Muudab eelmise valiku vastupidiseks.



Ääred ja servad. Selles näites on objektile lisatud teine materjal, et värvida servad punaseks.

Crease (viik)

Inner

Lisa viik sisemistele servadele.

Outer

Lisa viik välimistele servadele.

Rim

Lisa viik ääreservadele.

Even Thickness (ühtlane paksus)

Hoia paksus ühtlane, võttes arvesse teravaid nurki. Mõnikord parandab tulemust, kuid muudab arvutused aeglasemaks.

High Quality Normals (kvaliteetsed normaaliid)

Arvutab normaaliid ümber, et muuta paksus veel ühtlasemaks. Mõnikord parandab tulemust, kuid muudab arvutused aeglasemaks.

Fill Rim (täida äär)

Täidab ääre välimiste ja sisemiste servade vahel.

Rim Material (ääre materjal)

Kasuta ääre jaoks objekti teist materjali.

Subsurf Modifier - pinnatükeldaja

Mode: Kõik režiimid

Panel: Töötledjad (Modifiers)

Kirjeldus

Pinnatükeldus on meetod, mis jagab võre küljed osadeks, et anda kehale siledam väljanägemine, ja võimaldab modelleerida keerukaid ühtlasi pindu lihtsate, väheste tippudega võrede abil. Selle abil saab modelleerida kõrge tihedusega objekte, ilma et oleks vaja salvestada väga suurt hulka andmeid. See annab objektidele sileda, *orgaanilise* väljanägemise. Blender suudab nii modelleerimis- kui ka renderdusrežiimis arvutada jooksvalt igast tavapärasest võrest sileda jagatud külgedega võre, kasutades kas lihtsat pinna tükeldamist (Subsurf) või märksa nutikamat Catmull-Clarki meetodit.

Valikud



Töötaja paneel

Subsurf (pinna tükeldaja) on [töötaja](#). Et seda võrele lisada, vajuta Add Modifier (lisa töötaja) ja vali nimekirjast Subdivision Surface (pinnatükeldaja).

Catmull-Clark/Simple

Võimaldab valida ühe kahest tükeldusalgoritmist:

- Simple (lihtne) – Jaga ainult pinnad ilma neid silumata (tööpõhimõttelt sarnaneb Number of Cuts (lõigete arv) määrangule W → Subdivide (jaga osadeks) muutmisrežiimis (Edit Mode)). Sellest ei ole üldjuhul eriti kasu!
- Catmull-Clark – Vaikimisi väärtus, tükeldab ja silub pinnad.

Subdivisions (lõiketasemeid)

Määrab jagatud võre detailsuse 3D-vaates ja renderdamisel.

Märkus

Nende kahe valiku abil saab objekti manipuleerimisel 3D-aknas kasutada kiiret ja vähem ressursi vajavat lähendust lõplikule võrele ja renderdamisel palju kõrgema kvaliteediga versiooni.

Options (valikud)

- Optimal Display (optimeeritud esitus): Näitab traatvõre režiimis ainult algse võre servi, mitte tükeldatud tulemust, et anda parem ülevaade objekti kujust.
- Subdivide UVs (jaga UV-d): Kui see on sisse lülitatud, tükeldatakse ka UV-kaardid (st Blender lisab kõigile selle töötlejaga loodud osakülgedele "virtuaalsed" UV-koordinaadid).

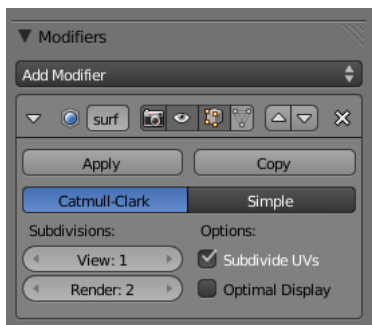
Et võre muutmisel näha ja muuta tükeldamise tulemust ("isooni"), peab lülitama sisse Editing Cage (redaktsioonivõre) režiimi, vajutades halli kolmnurga pildiga nupule töötaja paneeli päises (töötajat pinus üles ja alla liigutavate noolte kõrval). See võimaldab haarata punktidest nii, nagu nad asuvad uuel, tükeldatud pinnal, mitte esialgses võres.

Pinnatükeldus (*Subdivision Surfaces*)

Kirjeldus

Pinnatükeldus on meetod, kus pind jagatakse väiksemateks tükeldamiseks, et anda talle sujuvamat välimust. Ka väheste tippudega võre võib pärast pinnatükeldamist sujuv ja sile välja paista. Nii saab lihtsa vaeva ja väheste tippude arvuga modelleerida ümaraid orgaanilise välimusega objekte. Blender suudab pinnatükeldamist teha lennult nii muutmisrežiimis kui ka renderdamisel. Pane tähele, et pinnatükeldamine suurendab renderdamisel nõutava mälu mahtu ja võib seega renderdamist aeglustada.

Valikud



Pinnatükelduse töötaja (*Subdivision Surface Modifier*)

Pinnatükeldus (Subdivision Surface) on [töötaja](#). Selle võrele lisamiseks vali töötajate paneel Modifiers, vajuta töötajate lisamine (Add Modifier) ja siis vali nimekirjast Generate→Subdivision Surface (loo pinnatükelduse töötaja).

Type (tüüp)

Selle nupuga saab valida algoritmi:

- Catmull-Clark – vaikimisi variant, mis jaotab pinna väiksemateks tükideks ja muudab seda sujuvamaks.
- Simple (lihtne) – see jaotab pinna lihtsalt väiksemateks tükideks ilma sujuvamaks muutmata (sama nagu muutmisrežiimis tükeldamise tasemete (Levels) arv $W \rightarrow$ Subdivide). Selle kasulikkus ilmneb harva!

View (vaade)

See määrab resolutsiooni ehk jaotuste arvu *Blenderi interaktiivses 3D-vaates kuvamisel*.

Render (renderdus)

Seda jaotuste arvu kasutatakse *renderdamisel*.

Märkus

Nende kahe sätte abil saad kolmemõõtmeliste objektidega tegelemise jaoks kasutada kiiret ja kergekaalulist ligilähedast mudelit, kuid kasutada renderdamisel kõrgema kvaliteediga versiooni.

Subdivide UVs (tükelda UV-laotus)

Selle sisse lülitamisel tükeldatakse ka UV-laotused (st Blender lisab "virtuaalsed" koordinaadid kõikidele töötleja poolt tükeldatud külgedele).

Optimal Display (optimaalne kuvamine)

Visualiseerimise lihtsustamiseks saab selle nupu abil kuvatavaks jätta vaid algsed tükeldamata võred.

Viikide teravused pinna jaotamisel (*Weighted creases*)

Mode: Muutmisrežiim Edit, võre

Panel: 3D View \rightarrow Transform Properties

Menu: Mesh \rightarrow Edges \rightarrow Crease Subsurf

Kirjeldus

Tükeldatud pindade kaaluga serva viikide abil saad määrata seda, kuidas pinnatükeldamine gemoeetriat jagab, andes servadele kas sujuva või terava välimuse

Valikud

Valitud servade viigi kaalu määramine käib teisendamise omaduste alt (Transform Properties, N), kus tuleb kasutada keskmiseteisenduse liugurit (*Median Transform*). Kõrgem väärtus teeb serva "tugevamaks" ja tükeldamisel vastupidavamaks. Teine moodus selle meelepidamiseks on pidada seda kaalu serva teravuseks. Suurema kaalgua servasid moonutatakse tükeldamisel vähem. Tuleta meelde, et tükeldatud pind on kõigi ristuvate servade summa. Kui soovid mõne ala servasid teravamaks teha, pead suurendama ka kõigi ümbritsevate servade kaalu.

Armature Modifier - skeletitöötletaja

Mode: Object Mode (Objektirežiim)

Panel: Modifiers (Töötledjad)

Kirjeldus

Skelett (Armature) on luude süsteem, mille abil animeeritakse tegelaskujude ja muude objektide poose. Kasutades valikuid Vertex Group (tipugrupid) ja Multi Modifier (mitu töötletajat), on võimalik kasutada ühe objekti (võre) animeerimiseks mitut skeletti.

Skeleti lisamisega objektile saab objekti deformeerida ilma võre geomeetriat käsitsi muutmata. Skeletitöötletaja abil on võimalik objekti luude abil deformeerida, määraes lihtsalt skeleti nime, ilma et peaks kasutama (vana) "ülem/alam" süsteemi.

Skeleti kasutamise täpsemad juhised leiad peatükist [Taageldamine](#).

Valikud



Armature Modifier
(skeletitöötletaja)

Object (Objekt)

Skeletiobjekti nimi, mida see töötletaja kasutab.

Preserve Volume (säilita maht)

Kasuta kvaternioone, et interpolatsioonid oleks kenamad ja sujuvamad.

Vertex Group (tipugrupp)

Objekti selle tipugrupi nimi, mille kaalude järgi arvutatakse antud skeleti ja teiste skelettide koosmõju. Omab tähendust ainult siis, kui objektil on vähemalt kaks töötletajat ja Multi Modifier (mitu töötletajat) on sisse lülitatud.

MultiModifier (mitu töötletajat)

Kasuta samu sisendväärtusi, mis eelmine töötletaja (skelett?). See võimaldab objekti deformeerimiseks kasutada korraga mitut skeletti, mis kõik mõjuvad objekti "deformeerimata" kujule (st väldib seda, et teine skelett mõjuks esimese poolt juba deformeeritud objektile...). Skeletitöötletajate (Armature Modifier) mõjud arvutatakse kokku, kasutades kaalude määramiseks tipugruppe (Vertex Groups).

Bind To (seo)

Skeleti võre sidumise meetod(id).

Vertex Groups (tipugrupid)

Lülitab deformatsiooni määravad tipugrupid sisse/välja (st teatud nimega luud mõjuvad ainult neile tippudele, mis on sama nimega grupis).

Bone Envelopes (luuümbrikud)

Lülita deformatsiooni määravad "ümbrikud" sisse/välja (st luud mõjuvad neile tippudele, mis on ruumiliselt neile lähedal).

Invert (pööra ümber)

Pöörab eelmise seadega määratud tipugrupi mõju ümber (st muudab selle grupi kaalud vastupidiseks).

Cast Modifier - Lähendustöötleva

Mode: Kõik režiimid

Panel: Modifiers (Töötledjad)

Kirjeldus

See töötleva lähendab võre, kõvera, pinna või sõrestiku kuju mõnele eeldefineeritud geomeetrilisele kujule (kera, silinder, risttahukas).

Tulemus on sama, mis tööriista To Sphere (tee keraks) kasutamisel muutmiskontekstis (Editing Context) (Mesh (võre) → Transform (teisenda) → To Sphere Alt+ShiftS) või teiste programmide "keraks tegemine" (*Spherify*, *Spherize*) meetodi kasutamisel. Erinevalt nendest võimaldab antud töötleva aga kasutada ka teisi geomeetrilisi kujundeid peale kera.

Vihje

Cast Modifier on sageli praktiline koos [Silumistöötlevaga \(Smooth Modifier\)](#), et muuta lähendatud kuju kenamaks või parandada varjutuse ebakõlasid.

Oluline

Töökiiruse huvides kasutab see töötleva vaid objekti kohalikku koordinaatsüsteemi. Kui teisendatud objekti kuju ei ole õige, võib olla vajalik objekti pöörde rakendamine (CtrlA) - seda eriti juhul, kui lähendatakse silindrile.

Valikud



Cast Modifier
(lähendustöötleva)

Cast Type (lähendusviis)

Menüü, kust saab valida lähendusviisi (sihtmärkkuju): Sphere (kera), Cylinder (silindri) või Cuboid (risttahuka).

X, Y, Z

Lülitavad töötleva mõju X-, Y-, Z-telgede suunas sisse/välja (silindri korral ainult X ja Y suunas).

Factor (määr)

Määrab, kui palju ühtlustada algse kuju tippude asukohti sihtkujuga. Tegemist on lineaarse interpolatsiooni faktoriga: **0.0** jätab koordinaadid muutmata (st töötleva ei muuda midagi), **1.0** muudab kuju sihtmärgiga identseks. Väärtused vahemikus [0.0, 1.0] deformeervad võre; mõnikord väga huvitaval viisidel.

Radius (raadius)

Kui see on nullist erinev, määrab ta mõjuala ulatuse. Mõjualast väljaspool asuvaid tippu töötleva ei deformeeri.

Size (suurus)

Sihtmärkkuju suurus. Kui see on null, siis on suurus defineeritud algse kuju ja juhtobjekti poolt (kui see on olemas).

From radius (raadiusest)

Kui see on aktiivne, siis arvutatakse suurus (Size) raadiusest (Radius), et saavutada siledamat tulemust.

Vertex Group (tipugrupp)

Kui see on valitud, siis mõjub töötleva ainult nendele tippudele, mis asuvad vastavas grupis. Tipugrupi abil on võimalik võre tippu kaaludes (*vertex paint*) objekti reaajas ja soovitud viisil sihtmärkkujule lähendada.

Control Object (juhtobjekt)

Objekti nimi, mis juhib töötleva mõju. Selle objekti keskpunkt määrab sihtmärkkuju keskpunkti. Suurus ja pööre juhivad tippude teisendamist sihtmärkkujule. Vihje: juhtobjekti animeerimine (võtmeaadritega) animeerib ka töödeldud objekti.

Curve Modifier - kõveratöötleja

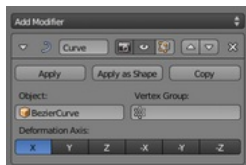
Mode: Object Mode (Objektirežiim)

Panel: Modifiers (Töötlejad)

Kirjeldus

Kõveratöötleja (Curve Modifier) käitub nagu tema [eelkäija](#) selle vahega, et enam ei ole vaja määratleda ülem/alam seost kõvera ja deformeeritava objekti vahel ja töötlejat saab rakendada kõigile objektitüüpidele reaalsajas.

Valikud



Curve Modifier
(kõveratöötleja)

Object (objekt)

Deformeeritavat objekti mõjutava kõveraobjekti nimi.

Vertex Group (tipugrupp)

Deformeeritava objekti tipugrupp. Töötleja deformeerib ainult tippe, mis asuvad selles grupis.

Deformation Axis (deformatsioonitelg)

X, Y, Z, -X, -Y, -Z: telg, mida mõõda kõvera deformatsioon mõjub.

Displace Modifier - nihutustöötaja

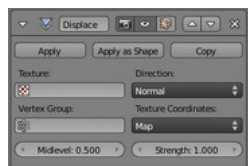
Mode: Kõik režiimid

Panel: Modifiers (Töötledjad)

Kirjeldus

Nihutustöötaja Displace Modifier nihutab tekstuuri intensiivsusest sõltuvalt võre tippe. See võimaldab kasutada nii protseduurilisi kui ka pilditekstuure. Nihutamine võib toimuda määratud kohaliku koordinaattelje suunas, piki tipunormaali või siis määravad tekstuuri R, G ja B komponendid tipu nihke vastavalt X, Y ja Z suunas.

Valikud



Displace Modifier
(nihutustöötaja)

Texture (tekstuur)

Tekstuuri nimi, millelt nihke väärtusi loetakse.
Kui see väli on tühi, on töötaja välja lülitatud.

Vertex Group (tipugrupp)

Rakendab töötajat ainult sellesse gruppi kuuluvatele tippudele.
Kui väli Vertex Group on tühi, rakendatakse töötajat ühesugusel määral kõigile tippudele.

Midlevel (kesktase)

Tekstuuri väärtus, mida loetakse neutraalseks (st nihe puudub). Keskvärtusest väiksemad tekstuuri väärtused tekitavad negatiivse nihke piki vastavat telge, suuremad väärtused positiivse nihke. Nihke suuruse valem on: $nihe = (tekstuuri\ väärtus) - keskvärtus$.
Tuleta meelde, et värvi ja intensiivsuse väärtused Blenderi tekstuuridel on vahemikus **0,0** kuni **1,0**, mitte **0** kuni **255**.

Direction (suund)

Suund, milles tippe nihutatakse.
Võib olla üks järgnevatest:

- X – nihuta piki X-telge.
- Y – nihuta piki Y-telge.
- Z – nihuta piki Z-telge.
- RGB -> XYZ – nihuta piki kohaliku koordinaatsüsteemi X-, Y- ja Z-telgi vastavalt tekstuuri R, G ja B (punane, roheline, sinine) komponentide väärtustele.
- Normal (normaal) – nihuta piki tipunormaali.

Texture Coordinates (tekstuurikoordinaadid)

Tekstuurikoordinaatide süsteem, mida kasutatakse igale tipule vastava tekstuuri väärtuse leidmiseks.
Võib olla üks järgnevatest:

- UV – kasuta külje UV-koordinaate.

UV Layer (UV-kiht)

UV-koordinaatide kiht, millelt lugeda tekstuurikoordinaadid.
Kui objektil ei ole UV-koordinaate, kasutatakse objekti lokaalset (Local) koordinaatsüsteemi. Kui see väli on tühi, aga objektil on UV-koordinaatide kiht (näiteks pärast võre esimese UV-kihi määramist), kirjutatakse väli üle parajasti aktiivse UV-kihiga.

Märkus

Kuna UV-koordinaadid on määratud iga külje (*face*) jaoks eraldi, omistatakse igale tipule selle külje vastav UV-koordinaat, millel teda esimesena kohati. Kõigile teistele külgedele, millel see tipp esineb, vastavaid UV-koordinaate ignoreeritakse. See võib tekitada moonutusi, kui võre UV-koordinaadid ei ole pidevad.

- Object (objekt) – loe tekstuurikoordinaadid teise objekti koordinaatsüsteemist (määratud väljal Object (objekt)).

Object (objekt)

selle objekti nimi, millelt tekstuurikoordinaadid loetakse. Objekti nihutamine muudab seega tekstuuri laotuse (*texture mapping*) koordinaate. Pane tähele, et algse objekti nihutamine muudab samuti tekstuurikoordinaate. Seega, kui tahad objekti, millele nihe on rakendatud, liigutades säilitada pinnanihke koordinaate, tuleb teist objekti liigutada täpselt sama palju samas suunas.
Kui väli on tühi, kasutatakse lokaalset koordinaatsüsteemi.

- Global (globaalne) – tekstuurikoordinaadid loetakse tippude globaalsetest koordinaatidest.
- Local (kohalik) – tekstuurikoordinaadid loetakse tippude lokaalsetest koordinaatidest.

Strength (tugevus)

Nihutamise määr. Lõplik tipu nihe saadakse nihke suuruse (pärast Midlevel (keskväärtuse) väärtuse lahutamist) korrutamisel nihke tugevusega (väli Strength). Valem on: $(tipu\ nihe) = (nihe) \times tugevus$.
Negatiivne tugevuse väärtus vahetab nihke suuna vastupidiseks.

Vaata ka

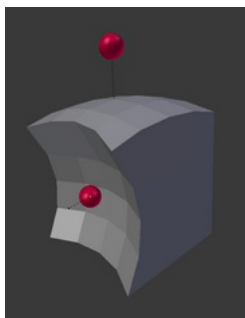
- Blender artists postitus: [Õppetükk Nihutaja kohta](#) (September 2006)

Hook Modifier - haaktöötleja

Mode: Kõik režiimid

Panel: Modifiers (Töötlejad)

Kirjeldus

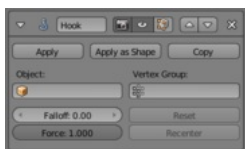


Kaks sfääri, mida kasutatakse haakidena (*hooks*), et deformeerida tükeldatud (*subdivided*) kuupi.

Haaktöötlejat (Hook Modifier) kasutatakse koos võtmevormidega (Shape Keys), et juhtida võre (Mesh) või sõrestiku (Lattice) deformatsioone teiste objektide abil (tavaliselt tühiobjektid (Empty), aga need võivad olla suvalised objektid).

Kui haak liigub, tõmbab ta võre tippe kaaludega määratud viisil kaasa. Kui oled varem kasutanud proportsionaalset muutmist, võid kujutada selle töötleja mõju ette nagu animeeritud proportsionaalset muutmist. Kuigi haagid ei võimalda tippude liikumist juhtida sama täpselt kui võtmevormid, on neid palju lihtsam kasutada.

Valikud



Hook Modifier
(haaktöötleja)

Object (objekt)
Haagina töötav objekt.

Falloff (vaibumine)
Kui nullist erinev, siis vahemaa, millest alates haagi mõju tipule puudub. Vaibumises kasutatakse sujuvat interpolatsiooni, mis on analoogne [proportsionaalse muutmisega](#).

Force (jõud)
Võimaldab määrata eri haakide mõju kaale, kui mitu haaki mõjuvad korraga samale tipule. Kaalumisreeglid on:

- Kui kõikide jõudude summa on väiksem kui **1,0**, siis ülejääk ($1,0 - (\text{jõudude summa})$) on jõud, millega mõjub tipu algne positsioon (nagu ta oleks üks haakidest).
- Kui kõikide jõudude summa on suurem kui **1,0**, mõjuvad ainult määratud haagid, proportsionaalselt nende kaaludega.

Järgnevad valikud töötavad ainult muutmisrežiimis (Edit Mode):

Reset (nulli)

Nulli haagi nihe.

Recenter (tsentreeri)

Liiguta haagi kese kursori positsioonile.

Select (vali)


Vali haagi poolt mõjutatud tipud võrel.

Reassign (määra ümber)

Seab valitud tipud selle haagi mõju alla.

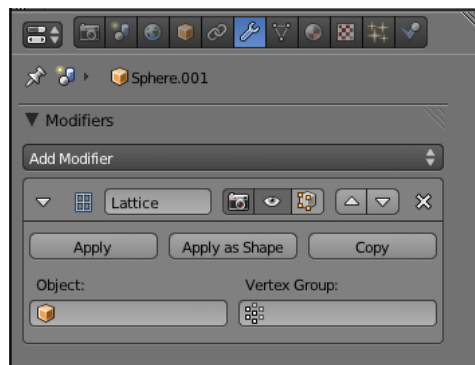
Lattice Modifier - sõrestiktöötleva

Mode: Object Mode (Objektirežiim)

Panel: *Properties Window* (seadistuste aken) -> Kontekstinupp Modifiers (töötlevad) 

Sõrestiktöötleva (Lattice Modifier) deformeerib baasobjekti vastavalt antud sõrestiku (Lattice) kujule.

Valikud



Lattice Modifier (sõrestiktöötleva).


Object (objekt)

Lattice (sõrestik), mille järgi baasobjekti deformeeritakse.

Vertex Group (tipugrupp)

Tipugrupi nimi. Kui see on määratud, mõjutab töötleva ainult sellesse gruppi kuuluvaid tippe.

Vihjed

Sõrestiku (Lattice) parameetreid saab muuta selle sõrestiku objektiandmete (Object Data) kontekstis omaduste aknas (Properties Window) .

Sõrestik koosneb kolmemõõtmelisest tippude ruudustikust, mis ei renderdu. Sõrestiku põhiline kasutus on tema poolt kontrollitavale objektile täpselt juhitud moonutuste lisamine (kas siis töötleva abil või ülem/alam suhte kaudu). Sihtobjektid võivad olla võred, kõverad, pinnad, tekstiobjektid, sõrestikud ja isegi osakesed.

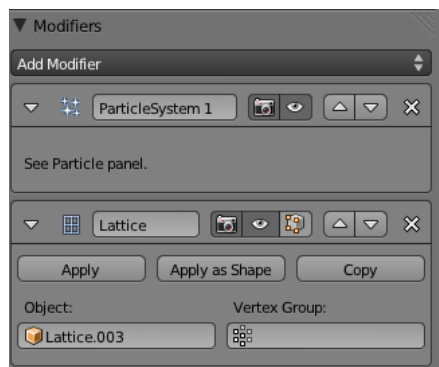
Miks kasutada sõrestikku objekti moonutamiseks, selle asemel et deformeerida objekti ennast muutmisrežiimis (Edit Mode)? Selleks on mitmeid põhjuseid:

- Esiteks: nii on lihtsam. Kuna võre võib sisaldada miljoneid tippe, võib kõigi nende valimine, skaleerimine ja nihutamine olla tüütu ettevõtmine. Kui sa aga kasutad selle asemel lihtsat sõrestikku, piirdub kogu töö mõne tipu nihutamisega.
- See on ilusam. Sõrestiku abil saavutatav deformatsioon näeb palju kenam välja!
- See on kiire! Sama sõrestiku abil saab deformeerida mitmeid võresid. Lihtsalt lisa igale võrele sõrestiktöötleva ja lase kõigil töötlevatel viidata samale sõrestikule.
- See on hea tava. Sõrestiku abil saab luua võrest uusi versioone minimaalse lisatöö ja ressursside kulutamisega. See aitab optimeerida stseeni ülesehitust, minimeerides modelleerimise mahtu. Sõrestik ei muuda võre pinna tekstuurikoordinaate. Sel viisil on lihtne teha väikesi muudatusi võreobjektidele ilma objekte endid muutmata.

Näited/Õppetükid

Blenderi versiooni 2.4 jaoks on õpetused [Õppetükide](#) peatükis. Üks versiooni 2.5 õppetükk näitab, kuidas [moodustada kahvlit](#).

Osakesed ja sõrestikud



Osakesed, mis järgivad sõrestikku.

Kui töötlevate järjekord on õige, järgivad osakesed sõrestikku. Osakeste süsteem peab asuma sõrestikust eespool!

Mesh Deform Modifier - võre moonutaja

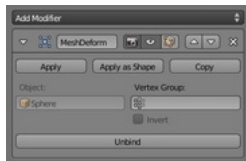
Mode: Kõik režiimid

Panel: Modifiers (Töötledjad)

Kirjeldus

Võre moonutaja (Mesh Deform Modifier) võimaldab kasutada suvalist suletud võret (ükskõik millise suletud vormiga, mitte ainult risttahukat, nagu (Lattice Modifier)) teist võret moonutava *puurina*.

Valikud



Mesh Deform Modifier (võre moonutaja)

Võre moonutajat on lihtne kasutada, kuid ta võib olla väga aeglane, sest moonutava võre ja moonutatava objekti vahelise korrektse vastavuse leidmine on arvutusmahukas.

Object (objekt)

Võre nimi, mida kasutatakse objekti moonutava *puurina*.

Vertex Group (tipugrupp)

Kui see on määratud, siis mõjub moonutus ainult sellesse gruppi kuuluvatele tippudele.

Invert (vaheta)

Pöörab eelmise määranguga seatud tipugrupi mõju ümber (st muudab selle grupi kaalud vastupidiseks).

Bind (seo)

Nupuga Bind (seo) luuakse võre moonutaja jaoks seos moonutava võre ja moonutatava objekti vahel nii, et moonutava võre kuju muutmine muudab objekti kuju.

Pane tähele, et sõltuvalt võre moonutaja (Mesh Deform Modifier) seadistustest ja moonutava võre ning moonutatava objekti keerukusest võib arvutustele kuluda märkimisväärselt aega. Sidumise ajal ei reageeri Blender kasutaja käsklustele. Samuti võib selle tegevuse käigus töömälu otsa lõppeda ja Blender kokku joosta.

Unbind (ühenda lahti)

Kui deformeeriv võre ja deformeeritav objekt on omavahel seotud, saab selle seose katkestada, vajutades nuppu Unbind, mis asendab nüüd paneelil nuppu Bind (seo).

Kui vajutada nuppu Unbind, säilitab moonutav võre oma hetkekuju ega pöördu tagasi algsele kujule enne moonutuse alustamist. Kui algset kuju on vaja säilitada, tuleb enne moonutuse alustamist teha moonutavast võrest koopia. Moonutatav objekt seevastu võtab tagasi oma algse kuju, mis tal oli enne moonutava võrega sidumist.

Precision (täpsus)

Väljal Precision saab määrata, millise täpsusega moonutav võre muudab moonutatavat objekti, kui võre punkte nihutatakse. Täpsuse välja võimalike väärtuste vahemik on **2** kuni **10**, vaikimisi väärtus on **5**. Selle väärtuse suurendamine tõstab oluliselt aega, mis võre moonutajal kulub sidumise arvutustele, kuid samas suurendab täpsust, millega moonutatav objekt järgib võre kuju muutusi. Arvutuste aja suurendamine võib panna Blenderi mitte alluma käsklustele, kuni ta on arvutused lõpetanud. Lisaks käsklustele mitteallumisele võib välja Precision suur väärtus ja väga keerukate (kas moonutava või moonutatava või mõlemate) objektide sidumine nõuda väga palju mälu ja äärmisel juhul Blenderi kokku jooksutada. Et kindel olla, salvesta oma töö enne võre moonutaja kasutamist!

See valik on nähtav ainult seni, kuni moonutav võre on sidumata.

Dynamic (dünaamiline)

Nupp Dynamic annab võre moonutajale märku, et tuleb arvestada ka deformeeritava objekti teisi moonutusi ja muudatusi lisaks moonutava võre muutmisele.

Kui nupp Dynamic on sisse lülitatud, arvestatakse moonutatava ja moonutava objekti sidumisel ka teisi võre kuju muutvaid meetodeid (nagu teised töötledjad ja võtmekujud). See muudab võre moonutuse täpsemaks. Vaikimisi on see välja lülitatud, et hoida kokku mälu ja arvutusvajadust objektide sidumisel.

Nagu ka Precision (täpsus), on see valik nähtav ainult seni, kuni moonutav võre on sidumata.

Vaata ka

- [Sõrestiktöötledja \(Lattice Modifier\)](#).

Shrinkwrap Modifier - sobitaja

Mode: Kõik režiimid

Panel: Modifiers (Töötledjad)

Kirjeldus

Sobitaja (Shrinkwrap Modifier) võimaldab objekti "kokku tõmmata" teise objekti pinnale. Ta liigutab modifitseeritava objekti iga tipu lähima punkti teise võre pinnal, kasutades ühte kolmest võimalikust meetodist. Sositajat on võimalik rakendada võredele, sõrestikele, kõveratele, pindadele ja tekstidele.

Nagu enamik teisi deformeerijaid, kasutab sobitaja "arvutatud" tippe, st objekti kehtivat geomeetria töötleva rakendamise hetkel, mitte algseid tippe või juhtpunkte.

Shrinkwrap on mõneti samane Blender 2.49 [retopo tööriistale](#), projitseerides tippe piki nende normaale või lähima punkti pinnal. Tal ei ole selliseid täpsusprobleeme nagu eelmisel versioonil, sest ta töötab objekti-, mitte pildi koordinaatsüsteemis. Samuti võimaldab ta "hoida vahemaa" sihtmärgist.

Märkus

Kui leiad, et Shrinkwrap Modifier on kasulik, kuid sooviksid sellega liigutada tühiobjekte või muuta teiste objektide asukohti, heida pilk [sobituspiirajale \(Shrinkwrap Constraint\)](#)!

Valikud

Target (sihtobjekt)

Sobitamise sihtobjekt, mille ümber objekti võre "mähitakse".

Vertex Group (tipugrupp)

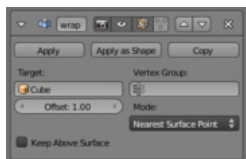
Selle grupi tippude kaalud määravad, kas ja kui palju iga tippu lõppasukoha suunas liigutatakse. Kui tipp ei ole grupi liige, siis ta jääb paigale (nagu kaalu 0 puhul).

Offset (vahemaa)

Vahemaa, mis jäetakse tipu ja väljaarvutatud sihtpositsiooni vahele Blenderi ühikutes.

Subsurf Levels (tükeldamistasemeid)

Rakendab objektile (või sihtmärgile?) enne sobitamist (ajutiselt) Catmull-Clark pinnatükelduse... Aga sellel ei näi hetkel olevat mingit mõju...



Nearest Surface Point (lähim pinnapunkt)

Mode (režiim)

See menüü määrab, kuidas arvutatakse lähim teise objekti pinna punkt iga esialgse objekti tipu jaoks. Mõned valikud lisavad paneelile täiendavaid spetsiifilisi välju.

Nearest Surface Point (lähim pinnapunkt)

Valib lähima punkti sihtmärgobjekti pinnal. Lisab täiendava valiku Above surface (pinna kohal), mis paigutab arvutatud tipud sihtmärgobjekti külgedest veidi eemale. Sel on tähendus ainult siis, kui Offset (vahemaa) ei ole null.



Project (projitseeri)

Projection (projektsioon)

See projitseerib tipud piki antud telge, kuni nad puudutavad sobituspinda. Tipud, mis projektsioonil ei löiku kunagi sihtmärgiga, jäetakse esialgsetele kohtadele. Seega, sõltuvalt selle valiku väärtusest ja mõlema objekti suhtelisest asendist, võib töödeldav objekt mõnikord jääda ka muutumatuks. See ei ole viga - proovi "mängida" valikutega (eriti Negative (negatiivse)/Positive (positiivsega)), või liiguta objekte ruumis...

Seda meetodit on kõige keerulisem edukalt kasutada, kuna mõnikord annab ta ootamatuid tulemusi. Ta lisab paneelile mõned spetsiifilised valikud:

X, Y, Z

Kohaliku koordinaatsüsteemi telg, piki mida projitseerimine toimub. Neid valikuid saab omavahel kombineerida, moodustades vastava "mediaan"-projektsioonitelge.

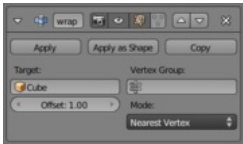
Negative (negatiivne), Positive (positiivne)

Võimaldab valida lubatud sobitamise suuna(d) piki projektsioonitelge. Kui kasutada rohkem kui ühte Shrinkwrap töötlevat, on võimalik kombineerida positiivseid ja negatiivseid projektsioonitelgi.

Cull Faces (eemalda küljed)

See võimaldab vältida projitseerimist sihtobjekti pinna "esiküljele" (või vastavalt "tagaküljele"). Külje "suuna" määrab

tema normaali suund (esikülg on see, "mille pinnalt" normaal "lähtub").
Auxiliary Target (lisaobjekt)
Täiendav objekt, millele projitseeritakse.



Nearest Vertex (lähim
tipp)

Nearest Vertex (lähim tipp)
See valib lähima tipu sihtobjektile. Sellel valikul ei ole täiendavaid välju.

Simple Deform Modifier - moonutaja

Mode: Kõik režiimid

Panel: Modifiers (Töötledjad)

Kirjeldus



Simple Deform
(moonutaja)

Moonutaja Simple Deform võimaldab objektidele (toetatud on võred, sõrestikud, kõverad, pinnad ja tekstid) vähese vaevaga mõned lihtsad moonutused lisada.

Nagu enamik teisi deformeerijaid, kasutab moonutaja "arvutatud" tippe ehk objekti kehtivat geomeetriat töötaja rakendamise hetkel, mitte algseid tippe või juhtpunkte. See tähendab, et moonutuse detailsust saab suurendada, lisades objektile (võrele) enne moonutajat Subsurface (pinnatükeldaja) või tõstes kõverate, pindade ja tekstide eelvaate lahtutust (Preview Resolution).

Kasutades teist objekti, on võimalik moonutuse telge ja lähtepunkti täpselt juhtida, luues väga mitmesuguseid efekte.

Valikud

Mode (režiim)

See menüü määrab, millist moonutusrežiimi neljast võimalikust kasutatakse:

- Twist (keera) – väärtab objekti ümber Z-telje.
- Bend (painuta) – Painutab objekti üle Z-telje.
- Taper (koonus) – Skaleerib objekti lineaarselt piki Z-telge.
- Stretch (venita) – Venitab objekti piki Z-telge (negatiivne Factor (faktor) väärtus surub teda kokku).

Vertex Group (tipugrupp)

Tipugrupi nimi, mis määrab, kui palju moonutus igat tippu mõjutab.

Origin (baas)

Objekti nimi, mis määrab deformatsiooni lähtekohta ja telgede suuna (tavaliselt tühiobjekt). Seda objekti võib:

- Pöörata, et muuta juhttelje suunda (sest selle objekt Z-telge kasutatakse moonutuse juhtteljena).
- Nihutada, et muuta moonutuse nullpunkti.
- Skaleerida, et muuta moonutuse faktorit.

Märkus

Kui baasobjekt (see, millele viidatakse väljal Origin (baas)) on moonutatava objekti alam, tekib Blenderi andmesüsteemis (DAG – "sõltuvusgraafik" (*dependency graph*)) tsükliiline sõltuvus. Seda probleemi saab lahendada, luues uue tühiobjekti ja liites sellele mõlemad objektid.

Factor (määr)

Moonutuse suurus. See võib olla ka negatiivne.

Limits (piirid)

See väli võimaldab määrata moonutuse alam- ja ülempiiri (tegemist on suhteliste ühikutega **0.0**-st kuni **1.0**-ni). Nagu arvata, ei saa ülempiir olla väiksem alampiirist.

Lock X Axis (lukusta X-telg)/Lock Y Axis (lukusta Y-telg) (Ainult Taper (koonus) ja Stretch (venita) režiimide puhul)

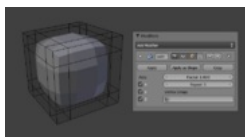
Määrab, kas tippude X- ja Y-koordinaadid muutuvad või mitte. Nii on võimalik moonutada objekti X-koordinaate, jättes Y-koordinaadid samal ajal muutumatuks.

Smooth Modifier - silumistöötleja

Mode: Kõik režiimid

Panel: Modifiers (Töötlejad)

Kirjeldus



Tükeldatud (*subsurfaced*)
kuubile rakendatud siluja



Nagu eelmine, kuid siluja
on rakendatud ainult ühele
tipugrupile

See töötleja võimaldab muuta võre siledamaks, vähendades nurki kõrvutiasetsevate külgede vahel, just nagu Smooth (silumine) muutmiskontekstis (Edit Context). Seega silub ta võret ilma seda jagamata: tippude arv jääb samaks.

Selle töötleja võimalused ei piirdu aga ainult silumisega. Silumise määra saab seada väljapoole **[0.0, 1.0]** piirkonda (kaasa arvatud negatiivsed väärtused), mis võib sõltuvalt algse võre kujust luua huvitavaid moonutusi.

Valikud

X, Y, Z

Võimaldavad lülitada töötleja mõju X-, Y- ja Z-telgede suunal sisse/välja.

Factor (määr)

Määrab silumise ulatuse. Mõistlikud väärtused on vahemikus **0.0**-st kuni **1.0**-ni (**0.0**: välja lülitatud, **0.5**: sama mis Smooth (silu) nupp, **1.0**: maksimaalne silumine). Väärtused väljapool antud piirkonda (rohkem kui **1.0** või vähem kui **0.0**) moonutavad võret.

Repeat (korda)

Silumisiteratsioonide arv - sama mis vajutada [Smooth](#) (silu) nuppu mitu korda.

Vertex Group (tipugrupp)

kui see on valitud, siis mõjub töötleja ainult nendele tippudele, mis asuvad vastavas grupis. Nii saab objekti interaktiivselt reaalajas siluda, maalides võre tippude kaale.

Warp Modifier - väändtöötleva

See deformeeriija võimaldab väga paindlikul viisil väänata või venitada võre osasid uude asukohta, määrares "alguse" ja "lõpu" piirkonnad kahe objektiga ja lastes kasutada ka mõjukõverat, tekstuuri ja tipugruppi.

Väänaja (Warp Modifier) võib alguses keeruline tunduda, kuid alljärgnev selgitus aitab aru saada, kuidas ta töötab. See töötleva vajab kaht punkti, mille määravad kahe objekti keskmed. "Alguspunkt" (*from*) märgib punkti ruumis, mida tõmmatakse "lõpp-punkti" (*to*) suunas. See sarnaneb [mõjualaga muutmisele \(Proportional Editing\)](#) muutmisrežiimis (Edit Mode).

Valikud

From (algus)

Määrab väände lähtepunkti koordinaatsüsteemi.

To (lõpp)

Määrab väände lõpu koordinaatsüsteemi.

Preserve Volume (säilita maht)

Säilitab ühte koordinaatsüsteemi pöörates objekti mahu.

Vertex Group (tipugrupp)

Moonutus mõjutab ainult antud gruppi kuuluvaid tippe.

Strength (tugevus)

Määrab, kui tugev väände mõju on.

Radius (raadius)

Määrab maksimaalse kauguse väände keskmeest, kus moonutus veel toimub.

Falloff Type (vaibumise viis)

Määrab, kuidas väände tugevus kahaneb moonutuse keskmeest raadiuse kauguseni. Vaata peatükist [mõjualaga muutmine](#) vaibumisviiside kirjeldusi.

Texture (tekstuur)

Määrab tekstuuri, mille väärtuse võrra jõudu muudetakse, et muuta nihe varieeruvaks.

Texture Coordinates (tekstuurikoordinaadid)

Määrab tekstuuri kasutamisel, kuidas see objektile laotatakse.

Object (objekt)

Kui tekstuurikoordinaatide allikaks on "Objekt", siis vastava objekti nimi.

UV Layer (UV kiht)

Kui tekstuurikoordinaatide allikaks on UV-kiht, siis vastava kihi number.

Wave Modifier - lainetöötaja

Mode: Object Mode (Objektirežiim)

Panel: Modifiers (Töötledjad)

Kirjeldus

Lainetöötaja (Wave Modifier) lisab objekti tippude/juhtpunktide Z-koordinaadile ookeanisarnase lainetuse. Seda töötajat saab rakendada võredele, sõrestikele, kõveratele, pindadele ja tekstidele, mõnede piirangutega *mitte-võre* objektide puhul:

- Valiku Normals (normaalid) aktiveerimine või nime sisetamine VGroup väljale deaktiveerib töötaja.
- Veel halvem - UV valimine tekstuurikoordinaatide allikaks jooksub Blenderi kokku!

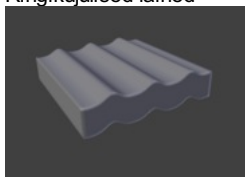
Valikud



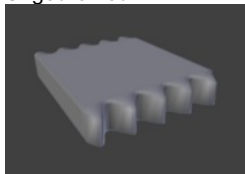
Wave Modifier
(lainetöötaja)



Ringikujulised lained



Sirged lained



Motion (liikumine) X ja
Normals (normaalid) Y
jaoks sisse lülitatud

Motion (liikumine)

X, Y, Cyclic (korduv): Laine deformeerib tippu/juhtpunkte Z suunas, lähtudes määratud punktist ja liikudes piki objekti ringikujuliste (X ja Y nupud mõlemad sisse lülitatud) või sirgjooneliste lainetena vastava telje suunas (kas X või Y nupp sisse lülitatud). Cyclic (korduv) kordab laineid tsükliliselt (mitte ei piirdu ühe lainega).

Normals (normaalid)

Ainult võrede jaoks. Liigutab võret piki pinnanormaale (mitte objekti Z-telge).

Time (aeg)

Aja juhtimisega seotud määrandud.

Offset (nihe)

Aja nihe kaadrites. Laine alguskaader (kui Speed (kiirus) on positiivne) või viimane kaader (kui Speed (kiirus) on negatiivne). Et lainetust enne animatsiooni algust käivitada, kasuta negatiivset alguskaadri väärtust.

Life (eluiga)

Laineanimatsiooni kestus kaadrites. Kui panna väärtuseks 0, kordub lainetus lõputult.

Damping (sumbumine)

Täiendavate kaadrite arv, mille jooksul laine sumbub algsest Height (kõrguse) väärtusest nullini pärast seda, kui tema eluiga (Life) on lõppenud. Sumbumine mõjub kõigile osalainetele ja algab esimesest kaadrist pärast eluea (life) lõppemist. Osalained kaovad sumbumise (Damping) väärtusega määratud arvu kaadrite jooksul.

Position (asukoht)

Lainete keskpunkti X ja Y koordinaadid objekti kohalikus koordinaatsüsteemis. Falloff (vaibumine) määrab, kui kiiresti laine vaibub, kui ta liigub üldtoodud koordinaatidega määratud lähtepunktist eemale. Pane tähele, et valik Start Position Object (alguspunkti objekt) elimineerib üldtoodud koordinaatide mõju, kuid säilitab vaibumise (Falloff) väärtuse.

Start Position Object (alguspunkti objekt)

Kasutab teist objekti lainete alguspunkti koordinaatidena. Jäta see väli tühjaks, kui soovid kasutada sisestatud koordinaate.

Pane tähele, et objekti kasutamisel saab selle asukohta animeerida, et muuta ajas lainete lähtepunkti.

Vertex Group (tipugrupp)

Ainult võrede puhul. Tipugrupp määrab, milline võre osa lainetusele allub ja kui palju (tippude kaalude järgi grupis).

Texture (tekstuur)

Tekstuuri kasutamine võimaldab täpselt määrata objekti tippude nihke suurust. Animeeritud tekstuurid võivad anda väga põnevaid tulemusi.

Texture Coordinates (tekstuurikoordinaadid)

Sellest menüüst saab valida meetodi, kuidas leitakse nihke tekstuurikoordinaadid:

Local (kohalik)

Objekti kohalik koordinaatsüsteem.

Global (globaalne)

Globaalne koordinaatsüsteem.

Object (objekt)

Lisab täiendava välja menüü alla, kust saab valida objekti, millelt tekstuurikoordinaate kasutada.

UV

Lisab UV-kihi (UV layer) menüü, kus saab valida objekti UV-kihi, millelt koordinaadid loetakse. **Hoiatus:** ära aktiveeri seda valikut objektide jaoks, mis ei ole võred, Blender paistab selle peale kokku jooksvat.

Speed (kiirus)

Lainete liikumiskiirus Blenderi ühikutes kaadri kohta.

Height (kõrgus)

Lainete algne kõrgus Blenderi ühikutes.

Width (laius)

Pool kaugusest (Blenderi ühikutes) kahe laineharja vahel (kui Cycle (kordus) on sisse lülitatud). See muudab kaudselt ka lainete amplituudi: kui lained on üksteisele liiga lähedal, ei jõua Z-nihke väärtus laineorgudes 0-ni. Sel juhul vähendab Blender kogu laine amplituudi, nii et miinimum on 0 ja maksimum vastavalt väiksem kui määratud väärtus. Vaata [tehnilisi detaile](#) allpool.

Narrowness (kitsus)

Iga laine tegelik laius - mida suurem on see väärtus, seda kitsam on laine. Ala tegelik laius, mille üks laine katab, on leitav valemiga $4/\text{kitsus}$. Seega, kui Narrowness (kitsus) väärtus on 1, on laine 4 ühikut lai, ja kui Narrowness (kitsus) on 4, on laine 1 ühik lai.

Hoiatus

Laine tegelike mõõtmete leidmiseks tuleb kõik siin toodud väärtused korrutada objekti skaalaga (Scale). Näiteks kui Scale Z (Z-skaala) väärtus on 2 ja laine kõrgus (Height) on 1, siis laine tegelik kõrgus on 2 **BU** (Blenderi ühikut)!

Tehnilised detailid ja soovitus

Siin kirjeldatakse ülalpool toodud parameetrite seost:



Lainefrondi omadused.

Et saada ilusaid, merelainetele ja sinusoidile sarnanevaid laineid, sea vahemaa kahe laine vahel võrdseks laine laiusega. St Narrow (kitsus) väärtus peab olema $2/\text{laius}$. Näiteks kui laius (width) = 1, pane kitsuse (narrow) väärtuseks 2.

Riide simulatsioon



Riide näidis



Riie nikerdatud puufiguuridel (autor motorsep).



Riide näidis

Riide simulatsioon on üks raskemaid arvutigraafika aspekte, kuna riie on petlikult lihtne reaalse maailma objekt, mida võetakse enestmõistetavana, kuid millel on tegelikult väga keerulised vastasmõjud iseenda ja keskkonnaga. Pärast aastepikkust arendamist on Blenderil väga töökindel riide simulaator, mida kasutatakse rõivaste, lippude, vimplite jms jaoks. Riidet mõjutavad liikuvad objektid, tuul ja teised jõud, samuti stseeni üldine aerodünaamiline mudel, mis kõik on sinu kontrolli all.

Kirjeldus

Riidetükk võib olla igasugune võre, avatud või suletud, mis on riidena määratud. Riide (Cloth) paneelid asuvad füüsika (Physics) alamkontekstis ja koosnevad kolmest seadete paneelist. Riie on kas avatud või suletud kaalutu võre – st eeldatakse, et ühel riideobjektile on kogu ulatuses sama tihedus (sama mass pinnahüki kohta).

Riie modelleeritakse harilikult ruudustikvõre või kuubina, kuid ta võib olla ka näiteks kas või kaisukaru. Siiski, Blenderi [pehmete kehade süsteem](#) (*Soft Body*) pakub suletud võrede jaoks paremat simulatsiooni. Riie on spetsialiseeritud kangaste jaoks.

Kui objekt on määratud riidena, lisatakse tema töötlejate pinusse automaatselt [riidetöötleja](#) (*Cloth Modifier*). [Töötlejana](#) toimib ta koosmõjus teiste töötlejate, näiteks skeleti (Armature) ja silujaga (Smooth). Nendel juhtudel arvutatakse võre lõplik vorm vastavalt töötlejate järjestusele pinus. Näiteks peaks riide siluma alles *pärast* seda, kui töötleja on arvanud tema kuju.

Riidet saad muuta kahes kohas. F7 füüsika nuppudega, mille abil saab muuta riide omadusi, ja töötlejate pinus, kus saad muuta riidetöötleja omadusi, mis on seotud väljanägemise ja koosmõjuga teiste töötlejatega.

Sa saad riidetöötlejat rakendada (Apply) võre külmutamiseks või paigalelukustamiseks selles kaadris, mis eemaldab töötleja. Näiteks võid sa laotada lina laua kohale, lasta simulatsioonil töötada ning seejärel rakendada töötleja. Sel juhul kasutad sa simulaatorit, et säästa kõvasti modelleerimisaega.

Simulatsiooni tulemused salvestatakse vahemällu, nii ei pea võre vormi, mis on animatsioonis iga kaadri jaoks ühe korra arvutatud, enam uuesti arvutama. Kui simulatsioonis tehakse muudatusi, on kasutajal täielik kontroll vahemälu puhastamise ja simulatsiooni taasjooksutamise üle. Simulatsiooni esimest korda jooksutamine on täiesti automaatne ja eeltöötlus ega erinevad sammud ei katkesta tööd.

Riide vormi arvutamine igal kaadril on automaatne ja toimub taustaprotsessina: nii saad simulatsiooni arvutamise ajal tööd jätkata. Siiski nõuab see tegevus olulist protsessorivõimsust ja sõltuvalt sinu arvuti kiirusest ja simulatsiooni keerukusest võib protsessori koormamine muuta Blenderi märgatavalt uimasemaks.

Ära hüpka edasi

Kui sead riide simulatsiooni valmis, kuid Blender ei ole veel arvanud vorme kogu simulatsiooni kaadri vahemiku jaoks ja sa hüppad oma animatsioonis palju kaadreid edasi, võib juhtuda, et riide simulaator ei suuda arvutada või näidata täpset võre vormi sellel kaadril, kui ta ei ole enne arvanud vormi eelmis(t)es kaadri(t)es.

Töö järjekord

Üldine protsess riidega töötamisel on:

1. Modelleeri riideobjektile üldine algusvorm.
2. Määra objekt objektikonteksti Object füüsika alamkontekstis Physics "riidena" (F7).
3. Modelleeri teised suunavad objektid, mis riidet mõjutavad. Tee kindlaks, et suunav töötleja (*Deflection*) on töötlejate järjestuses viimane, pärast teisi võret moonutavaid töötlejaid.
4. Valgusta riie ja määra materialid ja tekstuurid, tee UV-lahtilõige, kui soovid.
5. Kui soovid, anna objektile osakesed, nagu näiteks pinnalt kerkiv aur.
6. Jooksuta simulatsioon ja sedista valikud, et saada rahuldav tulemus. Ajagraafiku akna nn makinupud on selle sammu jaoks suurepärased.
7. Soovi korral peata võre mingis simulatsiooni punktis, et anda talle uus vaike-algusvorm.
8. Tee võres kaaderhaaval väiksemaid muudatusi, et parandada väikseid rebendeid.

Riidesimulatsioonide loomine

Selles lõigus selgitatakse, kuidas valida seaded soovitud tulemuse saamiseks. Esiteks, lülita sisse Cloth (riie). Määra, mis sorti riidet simuleerid. Võid valida alustamiseks ühe eeldefineeritud riidesortidest.

Nagu näed, mida raskem on kangas, seda jäigem see on, seda vähem venib ja seda vähem mõjutab seda õhk.

Riide paneel

Presets (eelseadistused)

Sisaldab mitut eelvalmistatud riidenäidist ja laseb sul lisada enda omi.

Quality (kvaliteet)

Seadista simulatsiooni sammude arv kaadri kohta. Kõrgemad väärtused annavad tulemuseks parema kvaliteedi, kuid on aeglasemad.

Materjal

Mass (mass)

Riide materjali mass.

Structural (struktuurne)

Riide üldine jäikus.

Bending (paindumine)

Kortsude koefitsient. Kõrgem väärtus tekitab rohkem suuri kortse.

Sumbumine (Damping)

Spring (vedru)

Riide liikumiskiiruse sumbumine. Kõrgem = sujuvam, vähem lainetust.

Air (õhk)

Õhul on harilikult mõningane tihedus, mis aeglustab esemete allalangemist.

Kinnitamine (Pinning)



Riidesimulatsioon

Esimene asi, mida sa riide kinnitamiseks vajad, on [tipugrupid](#) (*Vertex Groups*). Nende tegemiseks on erinevaid võimalusi, sealhulgas mõju maalimise tööriist (Weight Paint), mis võimaldab maalida alad, mida soovid kinnitada (vaata käsiraamatust [mõju maalimise](#) osa).

Kui tipugrupid on loodud, on asjad edasi üsna lihtsad, sa ei pea tegema muud kui vajutama nuppu Pinning of cloth (riide kinnitamine) paneelil Cloth (riie) ja valida, millist tippude gruppi tahad kasutada ja kui suurt jäikust sa sellele tahad.

Stiffness (jäikus)

Sihtmärgi asukoha jäikus. Jäikuse võid jätta nii nagu ta on, vaikimisi väärtus 1 on sobiv.

Kokkupõrked

Reeglina riidetükk ei ripu niisama 3D-ruumis, vaid põrkub vastu teisi keskkonnas olevaid objekte. Et kindlustada simulatsiooni õige toimimine, on vaja mitu asja seadistada ja koos tööle panna:


1. Riide objekt peab olema määratud osalema kokkupõrgetes.
2. Võid (soovitavalt) lasta riidel põrkuda ka riide endaga.
3. Teised objektid peavad olema riide objektile jagatud kihtide *abil* nähtavad.
4. Teised objektid peavad olema võreobjektid.
5. Teised objektid võivad liikuda või olla moonutatud kolmandate objektide poolt (nagu skelett või võtmevorm).
6. Teistele võreobjektidele tuleb öelda, et nad mõjutaksid riideobjekti.
7. Tööfail (.blend) tuleb salvestada kataloogi, et simulatsiooni oleks võimalik salvestada.
8. Seejärel tuleb simulatsioon eeltöödelda (Bake). Simulaator arvutab riide kuju etteantud kaadrite vahemikus.
9. Seejärel saad muuta simulatsiooni tulemusi või teha osades kaadrites täpsustusi riidevõres.
10. Sa saad teha muudatusi keskkonnas või moonutada objekte ning seejärel taaskäivitada riidesimulatsiooni käesolevast kaadrist alates.

Kokkupõrke seaded

Riide kokkupõrke (Cloth Collision) paneel.

Nüüd pead sa ütleva riideobjektile (Cloth), et tahad selle osalemist põrgetes. Otsi riideobjekti jaoks üles paneel Cloth Collision (riide kokkupõrked), mis on näidatud paremal:

Enable Collisions (luba kokkupõrkeid)

Klõpsa sellel  , ütlema riideobjektile, et ta peab eest liikuma.

Quality (kvaliteet)

Üldine seade, et määrata, kui kvaliteetne simulatsioon on. Suurematel numbritel on see aeglasem, kuid kindlustab, et rebendeid ja riidest läbitungimisi esineb vähem.

Distance (vahemaa)

Kui teine objekt satub riidele nii lähedale (Blenderi ühikutes), hakkab simulatsioon riidet eest ära lükkama.

Repel (tõuka eemale)

Tõukumisjõud, mida rakendatakse, kui riide on lähedal pörkele.

Repel Distance (tõukumisvahemaa)

Maksimaalne vahemaa tõukumisjõu rakendamiseks. Peab olema suurem kui minimaalne vahemaa.

Friction (hõõrdumine)

Koefitsient, mis määrab, kui libe on riide teiste võreobjektidega pörkudes. Näiteks siidil on madalam hõõrdekoefitsient kui puuvillal.

Kokkupõrked iseenesega

Päris riide ei tungi iseendast läbi, seega harilikult tahad sa, et riide pörkuks ka enesega.

Enable Self Collisions (lülita pörked iseenesega sisse)

Klõpsa seda, ütlema riidele, et ta ei läbistaks iseennast. See lisab küll simulatsioonile arvutusaega, kuid kindlustab realistlikuma tulemuse. Lipul, mida vaadatakse eemalt, ei ole vaja seda sisse lülitada, kuid lähivaatel keebist või pluusist tegelase seljas on see vajalik.

Quality (kvaliteet)

Et suurendada enesepörke kvaliteeti, suurenda lihtsalt välja Quality väärtust ning simulatsioon arvutab rohkem enesepörke kihte. Pea ainult mees, et väljade Collision Quality (kokkupörke kvaliteet) ja Quality (kvaliteet) väärtused peaks olema vähemalt samad.





Distance (vahemaa)

Kui tekib probleeme, võid muuta ka välja Min Distance (minimaalne vahemaa) väärtust enesepörke jaoks. Parim väärtus on 0.75; kiirete asjade jaoks pane parem 1.0. Väärtus 0.5 on küllaltki riskantne (arvatavasti on palju läbitungimisi), kuid töötab kiiremini.

Regressioonitesti fail: [Riide enesepõrked](#).

Jagatud kihid

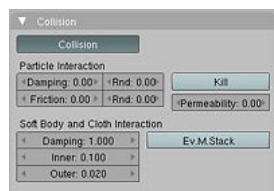
Oletame, et sul on kaks objekti: paar pükse kihitel 2 ja 3 ja tegelase võre kihitel 1 ja 2. Oled pükstel lülitanud eelpoolkirjeldatud moel sisse riidesimulatsiooni. Nüüd pead sa tegelase tegema riideobjektile "nähtavaks", nii et kui tegelane painutab jalga, tõukab see riidet. See põhimõte on sama kõikidel simulatsioonidel: simulatsioon toimib ainult objektidega, mis jagavad mõnda kihti. Selles näites jagavad mõlemad objektid kihti 2.

Et vaadata/muuta objekti kihte, selekteeri objekt, klõpsates RMB  3D-vaate objektirežiimis. Too kiirklahviga M esile hüpikaken "kihtide liigutamine" (*Move layers*), mis näitab kõiki kihte, kus objekt asub. Et asetada objekt ühele kihile, klõpsa LMB  vastava kihi nupul. Et asetada objekt mitmele kihile, klõpsa ⇧ Shift LMB  igal vastava kihi nupul. Et eemaldada objekt valitud kihilt, klõpsa lihtsalt uuesti ⇧ Shift LMB  vastava kihi nupul, et see välja lülitada.

Võreobjektide kokkupõrked

Kui pörkuv objekt ei ole võreobjekt, nagu näiteks NURBS-pind või tekstiobjekt, siis tuleb ta kõigepealt teisendada võreobjektiks. Et seda teha, selekteeri ta objektirežiimis ja vali 3D-vaate päises Object → Convert Object Type (objekt->konverteeri objektitüüp) (AltC) ning seejärel võta hüpikaknast Mesh (võre).

Riide/objekti pörked



Kokkupõrke seaded

Riideobjekti peab mõni teine objekt tõukama. Et riidet tõugata, peab sellel objektil olema sisse lülitatud kokkupõrge riidega. Et lülitada sisse riide/objekti pörked, pead sa suunamise sisse lülitama pörkuval objektile (mitte riideobjektile).

Otsi nuppude Buttons objektikonteksti Object füüsika alamkonteksti Physics paneelil küljel üles paneel Collision (kokkupõrked). Samuti on oluline tähele panna, et selle pörkepaneeli abil öeldakse kõikidele simulatsioonidele, et see objekt on osaline pörgetes ja teiste objektide mõjutamisel jagatud kihtide kaudu (osakesed, pehmed kehad ja riide).

Pane tähele

Kokkupõrgete (Collision) paneeli on kolm erinevat, mis kõik asuvad füüsika (Physics) alamkontekstis. Vaikimisi on esimene neist kaart väljade (Fields) paneeli kõrval, ja seda meil vaja ongi. Teine paneel, kaart pehme keha (Soft Body) grupis, käib pehmete kehade kohta (riidega ei ole sellel mingit tegemist). Ja viimast oleme me juba näinud: vaikimisi on see kaart riide (Cloth) paneeli kõrval.

Võreobjekti töötlejate pinu



Kokkupõrgete pinu

Objekti kuju deformeerib riid, seega peab riide simulatsioon teadma võreobjekti "tegelikku" kuju antud kaadris. See tõeline kuju on põhivorm, mida on muudetud võtmevormide või skelettide poolt. Seetõttu peab kokkupõrke (Collision) töötleja tulema **pärast** kõiki neid teisi töötlejaid. Pilt paremal näitab tegelase võreobjekti (mitte riideobjekti) töötlejate (Modifiers) paneeli.

Riide vahemälu

Vahemälu seaded riide jaoks on samad mis teistel dünaamilistel süsteemidel. Detailidest loe peatükist [Osakeste vahemälu](#).

Põrke eeltöötlus




Pärast seda, kui oled seadistanud suunava võre kaadrite vahemikus, mille vahel kavatsed simulatsiooni jooksutada (kaasa arvatud võre animeerimine skeleti abil), võid käskida riide simulatsioonil arvutada (ja vältida) kokkupõrked. Vali objektikonteksti füüsika alamkontekstis riideobjekt, seadista alguse (Start) ja lõpu (End) kaadrid, mille vahel tahad simulatsiooni arvutada, ja klõpsa nuppu Bake (eeltöötlus).

Sa ei saa muuta alguse ega lõpu väärtust, ilma et tühjendaksid simulatsiooni eeltöötluse. Kui simulatsioon on lõpetatud, märkad sa, et sul on valikud: vabasta eeltöötlus, muuda eeltöötlust või eeltöötle uuesti.

On mõned asjad, mida sa arvatavasti kohe tähele paned. Esiteks eeltöötleb ta märgatavalt aeglasemalt kui enne ja lõikub sageli võrega.

Vahemälu oleva simulatsiooni muutmine

Vahemälu sisaldab võre kuju igas kaadris. Sa saad muuta vahemälu olevat simulatsiooni, kui vajutasid eeltöötluse käivitamise järel nuppu Bake Editing (eeltöötluse muutmine). Mine lihtsalt kaadriale, mida tahad parandada, ja sisene klahviga ⇄ Tab muutmisrežiimi (Edit mode). Seal saad liigutada tippu, kasutades kõiki Blenderi võre vormimise tööriistu. Kui väljud, säilitatakse võre vorm sellel animatsiooni kaadril. Kui tahad, et Blender jätkaks simulatsiooni, kasutades edasi uut vormi, klõpsa LMB  nupul Rebake from next Frame (eeltöötle uuesti järgmisest kaadrist) ja esita animatsioon. Sel juhul Blender jätkab simulatsiooni, kasutades uut võre kuju.

Muuda võret, et parandada väiksemad rebendid ja kohad, kust pörkuv objekt on riid läbistanud.

Kui sa lisad, kustutad, surud välja või eemaldad võrest tippu, siis arvestab Blender uut võret kui algset võre kuju animatsiooni "esimese kaadri" välja, asendades algse vormi, millega sa alustasid, kuni kaadri, kus sa olid, kui võret muutsid. Seega, kui sa muudad võre ülesehitust, siis lahkudes ⇄ Tab abil muutmisrežiimist, peaksid sa tühjendama vahemälu, et Blender teeks kokkusobiva animatsiooni.

Probleemide lahendamine

Kui sul tekib probleeme pörketuvastamisega, on nende lahendamiseks kaks meetodit.

- Kiireim meetod on muuta minimaalse vahemaa (Min Distance) seadet riide kokkupõrke (Cloth Collision) paneelil. See on küll kiireim meetod läbistamise korrigeerimiseks, kuid on vähem täpne ja ei näe nii hea välja. Selle meetodi kasutamisel kipub riidest jääma mulje, nagu see ripuks õhus, üldse on üldmulje väga ümara väljanägemisega.
- Teine meetod on suurendada kvaliteeti (Quality) (esimesel riide (Cloth) paneelil). Selle tulemuseks on väiksemad sammud simulaatori jaoks ning seetõttu on suurem võimalus, et kiiresti liikuvad pörked registreeritakse. Samuti võid suurendada pörke kvaliteeti (Collision Quality), et pörke lahendamisel tehtaks rohkem iteratsioone.
- Kui ükski nendest meetoditest ei aita, saad hiljem hõlpsasti eeltöödeldud tulemust muutmisrežiimis (Edit mode) parandada.
- Moonutav võre rebestab minu riid: suurenda struktuuraset jäikust (väli StructStiff riide (Cloth) paneelil) väga kõrge väärtuseni, näiteks 1000.

Subsurf ehk pinnatükeldaja

Eeltöötlus tehakse iga pinnatükeldamise taseme kohta, seega kasuta nii renderdamisel kui ka eelvaatel **sama** pinnatükelduse taset.

Näited

Et riidega alustada, on esimese asjana vaja muidugi kangast. Seega kustutame vaikselt loodud kuubi ja lisame tasapinna. Mina suurendasin enda oma mõõda Y-telge, kuid see ei ole vajalik. Et saada head lõtku ja elastset kangast, on vaja seda mitu korda tükeldada. Selle näite jaoks tegin ma seda 8 korda. Seega mine kiirklahviga ⇄ Tab muutmisrežiimi} vajuta W → Subdivide multi (tükelda mitmeks) ja pane selle väärtuseks 8.

Nüüd teeme sellest riide, minnes objektikonteksti (Object context) (F7) → Physics (füüsika) alamkonteksti. Keri alla, kuni näed riide (Cloth) paneeli, ja vajuta nuppu Cloth (riie). Ilmub palju seadeid, millest enamikku me praegu ignoreerime.

See on kõik, mis on vaja teha, et seada riie valmis animeerimiseks, aga kui vajutad nüüd kiirklahvi AltA, siis su suurepärane riie lihtsalt langeb väga ebasuurepäraselt. Seda me käsitleme järgmises kahes lõigus kinnitamisest ja pörkumisest.

Simulatsiooni kasutamine võre vormimiseks/voolimiseks

Sa võid igal hetkel rakendada (Apply) riide (Cloth) töötleja, et külmutada võre kuju sellel kaadril. Seejärel võid riide taas sisse lülitada, seades alguse ja lõpu kaadrid, millest alates simulatsiooni edasi jooksutada.

Teine näide ajastamisest on lipp. Määra lipp lihtsa ruudustikuna ja kinnita serv lipuvarda külge. Simuleeri umbes 50 kaadrit ja lipp langeb rippuvasse "puhkeasendisse". Rakenda riidetöötleja (Cloth modifier). Kui tahad, et lipp lehviks või muudmoodi liiguks, lülita see taas sisse nende kaadrite jaoks, mil ta on kaamera vaateväljas.

Riide silumine

Kui oled järelnud samm-sammult õpetusi eelmises osas, siis on su riie arvatavasti natuke nurgeline. Et muuta seda kenaks ja siledaks nagu pildil, on sul vaja rakendada silumistöötleja (Smooth) ja/või pinnatükeldaja (Subsurf) muutmiskonteksti töötlejate (Modifiers) paneelil (F9). Seejärel leia samas kontekstis paneel Links and Materials (seosed ja materjalid) (sama, mida sa kasutasid tipugruppide) ja vajuta nuppu Set Smooth (muuda siledaks).

Kui sa nüüd vajutad kiirklahvi AltA, hakkab asi päris ilus välja nägema, kas pole?

Riie skeletil

Riie, mida moonutab skelett ja lisaks veel pörkuv objekt: [Regressioonitesti .blend-fail](#).

Riide kasutamine pehmete kehadena



Riide kasutamine pehmete kehadena

Riie saab kasutada ka pehmete kehade simuleerimiseks. See ei ole küll riide põhieesmärk, kuid töötab sellegipoolest. Näidispiilt kasutab standardset kummimaterjali (Rubber) ilma seadeid muutmata: lihtsalt AltA.

Blenderi fail näidispiildi jaoks: [Riide kasutamine pehmete kehadena](#).

Riie ja tuul



Lipp, millele mõjub tuul

Regressioonitesti .blend-fail riide jaoks koos tuulega ja iseenesega pörgetega (samuti .blend fail ülaloleva pildi jaoks): [Riideest lipp, millele mõjub tuul ja pörked iseenesega](#).

Kokkupõrked

[Osakesed](#), [pehmed kehad](#) ja [riideobjektid](#) võivad põrkuda võreobjektidega. [Parved](#) püüavad vältida kokkupõrke (Collision) objekte.

- Kokkupõrke toimimiseks peavad objektid jagama vähemalt ühte ühist kihti.
- Sa võid toimet osakestele piirata osakeste grupi objektidega ([väljamõju](#)) paneelil *Field Weights*.
- Pehme kehade *suunamine* on väga keeruline ja nad kipuvad sageli põrkuvatest objektidest läbi tungima.
- Karvad (Hair) ignoreerivad suunavaid objekte (kuid sa võid need animeerida pehmete kehadena, mis võtavad põrget arvesse).

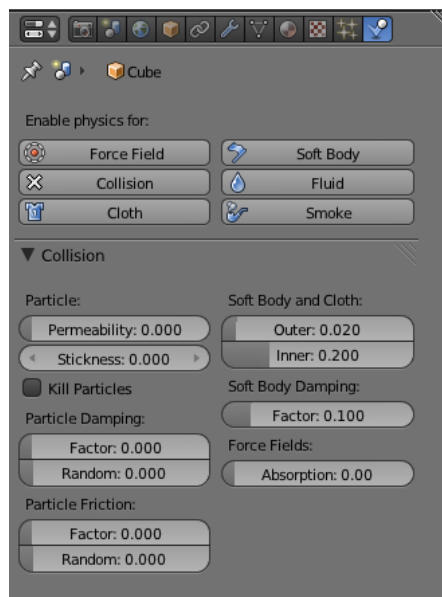
Kui muudad objektil põrke seadeid, pead sa uuesti arvutama osakeste, pehmete kehade või riide süsteemi (tühjenda vahemälu, Free Cache), see ei toimu automaatselt. Sa võid tühjendada vahemälu kõigil selekteeritud objektidel, valides CtrlB → Free cache selected (tühjenda valitud vahemälu).

Mode: Objektirežiim

Panel: Objekti kontekst Object → füüsika alamkontekst Physics → kokkupõrge (Collision)

Hotkey: F7

Valikud



Pilt 1: Füüsika alamkonteksti Physics kokkupõrke paneel Collision.

Permeability (läbistatavus)

Osa osakestest läbib võret. Seda saab animeerida objekti IPOdega kanalis Perm.

Stickiness (kleepuvus)

Kui palju osakesi jääb objekti külge kinni.

Kill Particles (tapa osakesed)

Kustutab osakesed kokkupõrkel.

Damping Factor (summutusfaktor)

Summutus põrke jooksul (ei sõltu osakeste kiirusest).

Random (juhuslik)

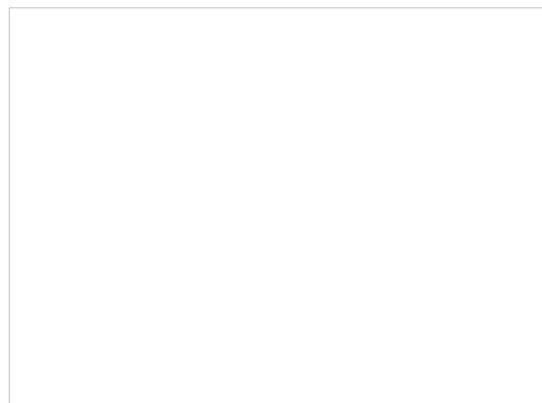
Summutuse juhuslik varieeruvus.

Friction Factor (hõõrdetegur)

Hõõrdumine liikumisel mööda pinda.

Random (juhuslik)

Hõõrdumise juhuslik varieeruvus.



Pilt 1b: Pehme keha tüüpi kokkupõrge tasandiga

Pehme keha ja riide vastastikune mõju

Outer (väline)

Välise pörketsooni suurus.

Inner (sisemine)

Sisemise pörketsooni suurus (vetrumise ulatus).

Välimine ja sisemine külg on määratud küljenormaaliga (*Pilt 1b*).

Damping Factor (summutustegur)

Summutus põrke jooksul.

*Pehme keha*de põrkeid on raske täiuslikuks saada. Kui üks objektidest liigub liiga kiiresti, tungib pehme keha võrest läbi. Vaata ka [pehmete kehade](#) peatükki.

Vastastikmõju jõuväljadega

Absorption (neeldumine)

Tõukaja mõjub ka mõjutajatele. Sa saad määrata mõned pörkuvaid/suunavad objektid, mis tõukavad kõrvale kindla osa mõjutaja jõust, neeldumise (Absorption) väärtuse abil. 100% neeldumise tulemuseks on, et jõud ei pääse läbi pörkuva/tõukava objekti. Kui sul on 3 pörkeobjekti, millel neeldumised on 10%, 43% ja 3%, annab see tulemuseks 50% neeldumist ($100 \times (1 - 0.1) \times (1 - 0.43) \times (1 - 0.03)$).

Näited



Pilt 2: Eemaletõugatud objektid

Siin on metaobjekt (Meta), mille tipud on duplitseeritud osakeste süsteemiks. Osakesi emiteeritakse allapoole ja kuupvõre tõukab neid eemale.

Vihjed

- Veendu, et võre pinna normaalid oleksid suunatud osakeste/punktide poole, et eemaletõukamine oleks õige.
- Karva (Hair) tüüpi osakesed reageerivad jõuväljadele otse, seega kui kasutad jõuvälja lühikeses ulatuses, ei ole põrget tingimata tarvis.
- Karva (Hair) tüüpi osakesed väldivad neid emiteerivat võret, kui muudad neid osakeste (Particle) režiimis. Seega saad vähemalt karvu põrkega modelleerida.

Explode Modifier - plahvatustöötleva

Mode: Kõik režiimid

Panel: Muutmiskontekst → Töötlevad (Modifiers)

Hotkey: None

Kirjeldus

Plahvatustöötleva (Explode Modifier) muudab võre geomeetriat (liigutades ja pöörates selle külgi(*face*)), jälgides (üldjoontes) vastava objektiga seotud osakeste trajektoore, tekitades efekti, nagu objekt plahvataks (ehk laguneks tükke, mis paiskuvad väljapoole).

Et plahvatus tekitaks nähtava efekti, tuleb ta rakendada võrele, millega on seotud osakeste süsteem - st see võre peab emiteerima osakesi. See osakeste süsteem määrab, kuidas võre plahvatab: ilma osakeste süsteemita võret see töötleva ei muuda. Plahvatuse detailsuse määravad nii süsteemis emiteeritavate osakeste hulk kui ka võre külgede arv. Kui kasutada vaikimisi seadistusi, siis mida suurem on osakeste ja külgede arv, seda detailsem on võre plahvatus, sest laialipaikuvaid tükke (külgi) on rohkem ja igat tükki (külge) juhitakse rohkemate osakeste poolt.

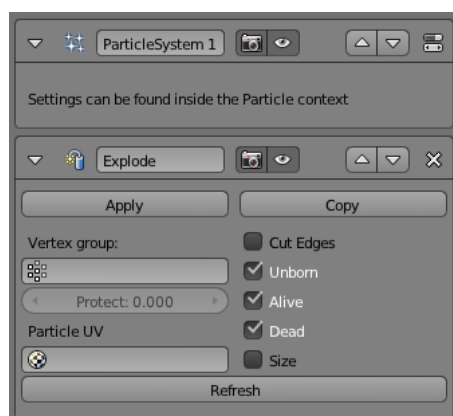
Siin on link Ogg Theora klipile, mis demonstreerib kuupi, millele on rakendatud osakeste süsteem ja plahvatustöötleva:

[Media:Manual - Explode Modifier - Exploding Cube - 2.5.ogg](#)

Siin on link plahvatava kuubi Blenderi failile. Et näha animatsiooni, vabastage osakeste puhver, vajutades Free Bake (tühjenda eeltöötlus) nuppu eeltöötuse (*Bake*) paneelil ja vajutage nuppu Animate (animeeri):

[Media:Manual - Explode Modifier - Exploding Cube - 2.5.blend](#)

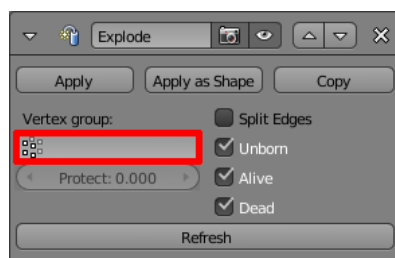
Valikud



Plahvatustöötleva (Explode Modifier) paneel osakeste süsteemiga (Particle System Modifier) selle kohal

Töötlevate järjestuse olulisus

Selle töötleva tulemus sõltub olulisel määral tema kohast töötlevate pinus. Kui ta asub osakeste süsteemist (*Particle System*) eespool, ei mõjuta osakesed teda ja ta ei muuda objekti. Osakeste süsteem (Particle System Modifier) peab asuma pinus enne plahvatust (Explode Modifier) sest viimane vajab plahvatuse juhtimiseks osakeste informatsiooni.



Plahvatustöötleva (*Explode Modifier*) paneel, kus *Protect this vertex group* (kaitse seda tipugruppi) väli on punasega esile toodud

Protect this vertex group (kaitse seda tipugruppi)

Kui plahvatustöötlevat kasutaval võrel on olemas tipugrupid, võimaldab see väli valida ühe neist. Seejärel kasutab plahvatus igale tipule mõjuva efekti suuruse arvutamiseks selle tipugrupi kaale. See võimaldab sõltuvalt kaalust kas jätta osad küljed plahvatuses täielikult muutmataks (kui vastava külje kaal on 1) või rakendab efekti täiel määral (kui vastava külje kaal on 0).

Vedeliku simulatsioon

Mode: Objektirežiim (Object Mode) / muutmisrežiim (Edit Mode) (võred)

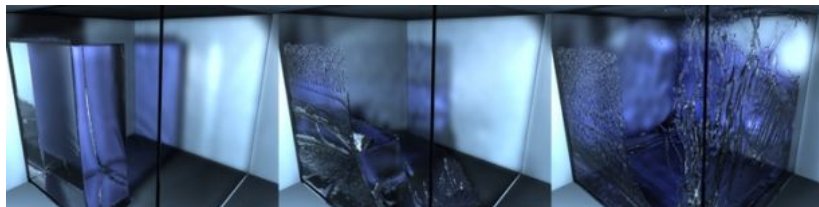
Panel: Füüsika (physics) allkontekst → Fluid (vedelik)

Hotkey: F7

Kirjeldus

Modelleerides Blenderis stseeni, saab määrata osad objektid võtma osa vedeliku simulatsioonist, näiteks kas siis vedeliku või takistusena. Teise objekti piirdkasti kasutatakse, et määrata risttahuka kujuline ala, milles simulatsioon toimub (nn "simulatsiooni toimumispiirkond"). Toimumispiirkonda määravale objektile saab seada üldised simulatsiooni parameetrid (nagu viskoossus ja gravitatsioon).

Kasutades eeltötluse (BAKE) nuppu, eksporditakse geomeetria ja seaded simulaatorisse ja teostatakse vedeliku simulatsioon, genereerides vedelikupinna võre koos eelvaatega iga animatsiooni kaadri jaoks, ja salvestatakse need kõvakettale. Siis laaditakse hetkel käsilolevale kaadrile vastav vedeliku pind kettalt ja kuvatakse või renderdatakse.



Puruneva tammi näide

Protsess

Üldiselt järgid sa järgmisi samme:

1. Modelleeri stseen (objektid, materjalid, valgus, kaamera).
2. Määra see osa stseenist, kus hakkab vedelik voolama (toimumispiirkond).
3. Täpsusta teiste vedelikuga seotud objektide funktsioonid (sissepääs, väljapääs, takistus jne).
4. Loo vedeliku allikas (allikad) ja täpsusta selle (nende) materjal, viskoossus ja algiirius.
5. Eeltötle esialgne simulatsioon.
6. Tee vajalikud muudatused ja salvesta.
7. Eeltötle lõplik simulatsioon.

Kui eeltötlus simulatsiooni, siis muutub versioonitunnus kasutajaeelistuste (User Preferences) akna päises olekuribaks, mis näitab eeltötluse edenemist. Eeltötlus nõuab **palju** arvutusvõimsust (ja aega). Seda ei ole mõtet jälgida, sest nagu öeldakse: vesi, mida jälgid, ei hakka keema, ja kook, mida vaatad, ei küpse; ja vedeliku eeltötluse vaatamine on nagu jälgida muru kasvamist. Eeltötlust on parim teha üleöö.

Stseenidevahelise ülekandmise ja linkimise võimalikkuse tõttu saab olla ühe .blend-faili kohta ainult üks toimumispiirkond.

Valikud



Blender 2.5 vedelikusimulatsiooni seaded, valitud on Domain (toimumispiirkond)

Domain (toimumispiirkond)/Fluid (vedelik)/Obstacle (takistus)/Inflow (sissevool)/Outflow (väljavool)/Particle (osake)/Control (juhtimine)
Nendest nuppudest ühe valimine määrab, mismoodi valitud objekti simulatsioonis kasutatakse. Igaüks nendest nuppudest määrab erineva funktsiooni ja vastavalt ka erinevad edasised valikud.

Võre renderdamine

Kui võrel on töötlejad, kasutatakse võre eksportimisel vedelikuarvutajasse renderdamisseadeid. Olenevalt seadetest võib arvutusaeg ja mälu kasutus plahvatuslikult kasvada. Näiteks kasutades liikuvat võret koos pinnatükeldajaga (Subsurf) takistusena, võib pinnatükelduse väljalülitamine või tasemete vähendamine aidata vähendada simulatsiooniaega. Kui seaded/liigendus on sellised nagu vaja, võid hiljem alati suurendada seadeid, et saavutada realistlikum tulemus.

Toimimispiirkond (*Domain*)

10 cm kruus lahutusega 70

10 cm kruus lahutusega 200

Objekti piirdkast määrab simulatsiooni piirid. **Kõik vedelikobjektid peavad olema toimimispiirkonna sees.** Sellest väljaspool olevaid vedelikobjekte ei eeltöödelda. Väljapoole toimimispiirkonda ei saa sattuda mitte ühtegi tilka, nähtamatud jõuväljad hoiaks nagu vedelikku 3D-ruumis. Stseenis saab olla ainult üks vedeliku simulatsiooni toimimispiirkonna objekt.

Ei ole vahet, mis kujuga on see objekt, sest seda koheldakse *alati* kui kasti (piirdkasti küljed võivad olla erineva pikkusega). Seega ei ole harilikult vaja kasutada muud keha kui kast. Kui vajad vedeliku voolu mõjutamiseks takistusi või teisi piire peale kasti, siis pead lisama täiendavaid takistusobjekte toimimispiirkonna piiride sisse.

See objekt "asendatakse" simulatsiooni käigus vedelikuga.

Lahutus (*Resolution*)

Detailsus, millega tegelik vedeliku simulatsioon läbi viiakse. See on simulatsiooni juures ilmselt kõige olulisem seade, kuna määrab vedeliku detailsuse, mälu ja ketta kasutuse ning samuti ka arvutusaja.

Pane tähele, et vajamineva mälu hulk kasvab kiiresti: lahutus 32 vajab umbes 4 MB, 64 umbes 30 MB, seevastu 128 vajab juba rohkem kui 230 MB. Olenevalt sellest, kui palju mälu sul on, veendu, et resolutsioon on piisavalt madal, et hoida Blenderit hangumast või kokku jooksmast. Pea meeles, et mitmetel operatsioonisüsteemidel on piiratud see, kui palju mälu ühele "protsessile", nagu näiteks Blenderile eraldatakse, isegi kui "masinal" on mälu sellest rohkem. Tee kindlaks, mis on sinu arvuti piirangud.



Toimimispiirkond maailmas

Veendu, et seadistad lahutuse parajaks toimimispiirkonna suuruse jaoks reaalses maailmas (Toimimispiirkonna tegelik füüsiline suurus meetrites määratakse vedeliku kastikese all asuval väljal *Domain World* (Toimimispiirkond maailmas)). Kui toimimispiirkond ei ole kuubikujuline, määratakse lahutus pikima külje järgi. Teiste külgede lahutust vähendatakse vastavalt nende pikkusele (seetõttu vajab sama resolutsiooniga mittekuubikujuline toimimispiirkond vähem mälu kui kuubikujuline).

Eelvaate lahutus (*Preview Res*)

See on lahutus, millega genereeritakse pinnavõrede eelvaade. Seega ei mõjuta see tegelikku simulatsiooni. Isegi kui eelvaates "ei ole midagi näha", võib seal olla õhuke vedelikupind, mida eelvaates ei ole võimalik näidata.

Alguse ja lõpu aeg

Start (algus) on simulatsiooni algusaeg (sekundites). See valik paneb simulatsiooni arvutamise Blenderis algama tegeliku simulatsiooni algusest hiljem. Toimimispiirkonna deformatsioone ja vedeliku voolamist, mis eelnevad algusajale, ei salvestata.

Näiteks kui tahad, et vedelik ilmuks alles siis, kui ta on enne tegelikku esimest kaadrit juba 4 sekundit voolanud, peaksid sisestama siia 4.0.

End (lõpp) on simulatsiooni lõpu aeg (sekundites).

Alguse ja lõpu aegadel ei ole mingit tegemist sellega, kui mitu kaadrit eeltöödeldakse - need põhinevad füüsikalisel jõul ja vedeliku viskoossusel.

Eeltöötlus algab alati kaadris 1:

Vedeliku simulaator eirab seadet Sta seadet animatsiooni paneelil ja eeltöötlus algab alati kaadrist 1.

Kui soovid, et simulatsioon algaks hiljem kui kaader 1, pead sa muutma vedelikobjektid oma toimispiirkonnas mitteaktiivseks

kuni kaadrini, millest alates sa tahad simulatsiooni alustada. Täpsema informatsiooni saamiseks loe [altpoolt](#).

Eeltöötlus lõpeb alati lõpukaadril End, mis on määratud animatsioonipaneelil (Anim):

Kui su kaadrisagedus on 25 kaadrit sekundis ja lõpu aeg on 4.0 sekundit, siis sa peaksid (kui alguse aeg on 0) seadma animatsiooni lõppema kaadril $4.0 \times 25 = 100$.

“Alguse aeg” ja “lõpu aeg” on väljendatud “simulatsiooniajas” (sekundites):

Kui sead oma algusajaks (Start) 3.0 ja lõpuajaks (End) 4.0, siis simuleerid sa vedeliku liikumist 1 sekundi jooksul. See üks sekund jaotatakse nii mitme kaadri vahel, nagu sa seadsid animatsioonipaneeli väljal End (lõpp). See tähendab, et kui sa näiteks oled Blenderi seadnud tegema 250 kaadrit kaadrisagedusel 25 kaadrit sekundis (Scene (stseenikontekstis) → Render alamkontekstis → paneelid Anim (animatsioon) ja Output (väljund)), paistab nagu oleks vedelik simulatsiooni alguseks juba kolm sekundit voolanud, *kuid* liigub edasi aeglustatult (üks kümnendik normaalsest kiirusest), sest kümnes sekundis videos on üks sekund simulatsiooni. Et seda parandada, muuda lõpu aeg $3.0 + 10.0 = 13.0$, et see vastaks 250 kaadrile sagedusel 25 kaadrit sekundis. Nüüd on simulatsioon reaajas, sest sa seadsid 10 sekundit vedeliku liikumise simulatsiooni 10 sekundile animatsioonile. Need seaded võimaldavad sul juhtida simulatsiooni “kiirust”.

Kuvamise kvaliteet (Disp.-Qual)

Kuidas kuvatakse eeltöödeldud simulatsioon Blenderi 3D-vaates (rippmenüü pealkirjaga *Viewport Display* (esitus vaates)) ja renderdamisel (rippmenüü pealkirjaga *Render Display* (esitus renderdamisel)):

- Geometry (geomeetria):: kasuta originaalgeomeetriat (enne simulatsiooni).
- Preview (eelvaade):: kasuta eelvaate võret.
- Final (lõplik):: Kasuta kõrglahutusega võret.

Kui eeltöötlemisandmeid ei leita, kuvatakse vaikimisi originaalvõre.

Pärast toimimispiirkonna eeltöötlemist kuvatakse see (harilikult) Blenderi aknas eelvaate võrena. Et näha toimimispiirkonna originaalsuurust ja ulatust, vali vasakust rippmenüüst Geometry (geomeetria).

Eeltötluse kataloog (Bake Directory)

VAJALIK! Kataloog ja failide eesliide, millega eeltööteldud pinnavõred salvestatakse. See on samane animatsiooni väljundseadetele, ainult faili valimine on natuke erinev: kui valid mõne eelnevalt loodud pindvõre (näiteks “untitled_OBcube_fluidsurface_final_0132.bobj.gz”), siis eesliiteks seatakse sel juhul automaatselt (“untitled_OBcube_”). Nii saab simulatsiooni teha erinevate seadetega ja see lubab kiireid muudatusi erinevate pinnaandmete seadete vahel.

Vaikeväärtus on “/tmp/”, mida sa ilmselt “ei” taha. Vali sobiv kataloogi nimi ja faili eesliide, nii et need failid salvestataks sobivasse kohta ja oleks nimetatud nii, et kaks erinevat vedeliku simulatsiooni ei kirjutaks üksteist üle (kui kavatsed määrata ainult kataloogi nime, näiteks ilma failinime eesliiteta, ära unusta lisada “/”).

Nupp “Bake” (eeltöötle)

Teostab tegeliku vedeliku simulatsiooni. Blender jätkab normaalselt tööd, kui välja arvata see, et akna üläääres, renderdamise rippmenüü kõrval näidatakse olekuriba. Vajutades kiirklahvi Esc või “x” olekuriba kõrval, saad simulatsiooni katkestada. Hiljem on iga kaadri jaoks valitud kataloogis kaks “.bobj.gz” faili (üks lõpliku (Final) kvaliteedi, üks eelvaate (Preview) kvaliteedi jaoks), pluss üks “.bvel.gz” (lõpliku (Final) kvaliteedi) fail.

Märkused...


Eeltöödeldud andmete vabastamine

“Bake” kataloogi sisu kustutamine on destruktiivne viis seda teha. Ole ettevaatlik, kui rohkem kui üks simulatsioon kasutab sama eeltötluse kataloogi (tee kindlaks, et neil on erinevad failinimed, muidu hakkavad nad üksteist üle kirjutama)!

Eeltötluste taaskasutamine

Eelnevalt salvestatud (eeltöödeldud) kataloogi ja failinime maski käsitsi sisestamine (või otsimine) vahetab vedeliku voolu ja võre moonutused nende vastu, mis olid selle vana eeltötluse ajal. Seega saad eeltöödeldud voogusid kasutada lihtsalt sellel väljal neile osutades.

Eeltöödeldud toimimispiirkonna valimine

Pärast toimimispiirkonna eeltöötlemist muutub see vedeliku võreks. Et toimimispiirkonda uuesti valida, nii et saaksid seda pärast tehtud muudatusi jälle eeltöödelda, mine mõnele kaadri ja selekteeri (RMB ) vedeliku võre. Siis võid klõpsata uuesti nuppu BAKE (eeltöötle), et arvutada uuesti vedeliku voolamine selles toimimispiirkonnas.

Teised toimimispiirkonna valikud

Domain World (toimimispiirkond maailmas)

Viscosity (viskoossus)

Vedeliku “paksus” - tegelikult jõud, mida on vaja, et kindla pindalaga objekti kindla kiirusega sellest läbi liigutada. Võid sisestada väärtuse käsitsi või kasutada mõnda rippmenüüs olevatest algseadistustest (nagu mesi, õli või vesi). Käsitsi sisestamisel pane tähele, et reaalses maailmas mõõdetakse viskoossust (nn dünaamiline viskoossus) paskal-sekundis (Pa.s), või Poise'i ühikutes (P, võrdne 0.1 Pa.s, hääldatakse “*puaz*”, nimetatud prantslase Jean-Louis Poiseuille järgi, kes avastas “viskoossete vedelike laminaarse voolamise” seadused), ja harilikult sentipoise'ides (cP, võrdne 0.001 Pa.s, “*sentipuz*”). Teisalt kasutab Blender aga kinemaatilist viskoossust (mis on dünaamiline viskoossus Pa.s, jagatud tihedusega kg.m^{-3} , mille ühik on $\text{m}^2 \cdot \text{s}^{-1}$). Allpool olev tabel annab mõned näited vedelikest koos dünaamilise ja

kinemaatilise viskoossusega.

Käsitsi sisestatud väärtused määratakse ujukomaarvu ja astendajaga. Need ujukomade ja astendajate jaoks mõeldud väljad (teaduslik tähistus) lihtsustavad väga väikeste või väga suurte numbrite sisestamist. Vee viskoossus toatemperatuuril on 1.002 cP või 0.001002 Pa.s; vee tihedus on umbes 1000 kg.m^{-3} , mis annab kinemaatiliseks viskoossuseks $0.000001002 \text{ m}^2.\text{s}^{-1}$ - seega sisestus peaks olema 1.002 korda 10 astmel miinus kuus (1.002×10^{-6} teaduslikus tähistuses). Tuline klaas ja sularaud on vedelikud, kuid väga paksud; nende jaoks peaksid sa kineetilise viskoossusena sisestama midagi sellist nagu 1.0×10^0 (= 1.0) (mis vastab umbes väärtusele 1.0×10^6 cP). Pane tähele, et simulaator ei sobi mittevedelike ehk "mittevoolavate" materjalide jaoks. Seades viskoossuse väga suurtele väärtustele, saame jäiga keha käitumise, kuid see võib põhjustada ebastabiilsust.

Viskoossus varieerub

Blenderi vaikeväärtused on tüüpilised seda sorti vedelikele ja paistavad animeerituna "õiged". Ent mõnede vedelike tegelik viskoossus, eriti suhkrurikaste vedelike, näiteks šokolaadisiirupi ja mee viskoossus sõltub suuresti temperatuurist ja kontsentratsioonist. Õli viskoossus varieerub SAE märgistuse järgi. Klaas toatemperatuuril on põhiliselt tahke, kuid 1500 kraadi juures voolab (peaaegu) nagu vesi.

Blenderi viskoossuse ühikute teisendus

Vedelik	dünaamiline viskoossus (cP)	kinemaatiline viskoossus (Blender, $\text{m}^2.\text{s}^{-1}$)
Vesi (20 °C)	1.002×10^0 (1.002)	1.002×10^{-6} (0.000001002)
Õli SAE 50	5.0×10^2 (500)	5.0×10^{-5} (0.00005)
Mesi (20 °C)	1.0×10^4 (10 000)	2.0×10^{-3} (0.002)
Šokolaadisiirup	3.0×10^4 (30 000)	3.0×10^{-3} (0.003)
Ketšup	1.0×10^5 (100 000)	1.0×10^{-1} (0.1)
Sulaklaas	1.0×10^{15}	1.0×10^0 (1.0)

Realworld-size (suurus reaalses maailmas)

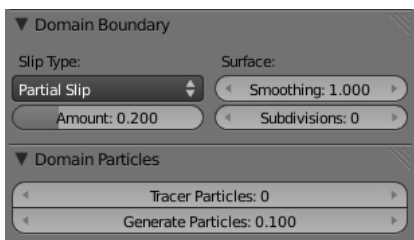
Toimimispiirkonna objekti suurus reaalses maailmas meetrites. Kui tahad luua tassi kohvi, võiks see olla 10 cm (0.1 meetrit), samas ujumisbassein võiks olla 10 m. Suurus seatakse siin toimimispiirkonna piirdkasti pikima külje järgi.

Gridlevel (ruudustiku tase)

Kui palju adaptiivseid ruudustiku tasemeid simulatsioonis kasutatakse - seades selle väärtuseks -1, on valik automaatne.

Compressibility (kokkusurutavus)

Kui sul probleeme suurte seisva vedeliku kogustega, millel on kõrge lahutus, siis võib aidata, kui seda numbrit vähendada (pane tähele, et see suurendab arvutusaega).



Teised vedeliku toimimispiirkonna valikud

Domain Boundary (toimimispiirkonna piirid)

Selles kastikeses on kõik libisemise ja pinna valikud.

Boundaries (piirid)

Boundary type (piiri tüüp)

See on sama nagu takistusobjektidel allpool, põhimõtteliselt seab see toimimispiirkonna kuus külge kas kleepuvaks, mittekleepuvaks või kusagile vahepeale (seda seatakse välja PartSlip Amount (osakeste libisemise määr) väärtusega).

Surface Smoothing (pinnasilumine)

Vedeliku pinnale rakendatava silumise määr. 1.0 on standardne, 0 on väljalülitatud, suuremad väärtused suurendavad silumist.

Subdivisions (tükeldustasemeid)

Lubab luua kõrgeresolutsioonilisi pindvõresid otse simulatsiooni käigus (vastupidiselt pinnatükeldamise töötlejale, mis teeb seda hiljem). Väärtus 1 tähendab, et pinnatükeldamist ei ole ja iga suurendamine annab juurde ühe pinnatükeldamise taseme iga vedeliku vokseli kohta. Tulemuseks saadav võre kasvab kiiresti mahukaks ja vajab palju kettaruumi. Ole ettevaatlik selle kombineerimisel suurte silumisväärtustega - see võib võre pinna genereerimisel viia pikkade arvutusaegadeni.

Generate SpeedVecs/Disable (Loo kiirusvektorid/lülita välja)

Kui seda nuppu klõpsata, siis kiirusvektoreid ei ekspordita. Seega vaikimisi luuakse kiirusvektorid ja salvestatakse kettale. Seda saab kasutada komposiitsõlmedega pildipõhise liikumishägu arvutamisel.

Domain Particles (toimimispiirkonna osakesed)

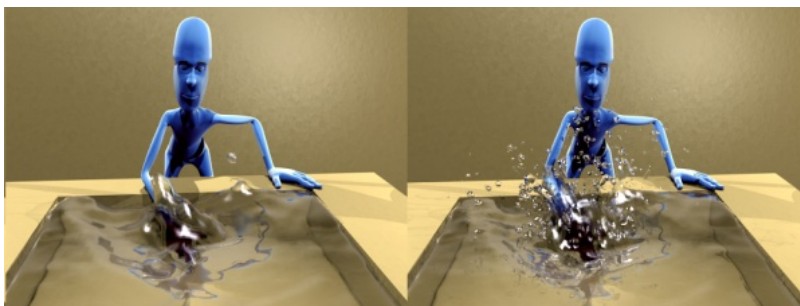
Siin saad vedeliku simulatsioonile lisada osakesi, et visuaalset efekti parandada.

Tracer Particles (jälgivad osakesed)

Simulatsiooni alguses vedelikku lisatavate järgivate osakeste arv. Et neid kuvada, loo teine objekt vedeliku tüübiga Particle (osake), mida on selgitatud allpool ja mis kasutab sama eeltöötuse kataloogi kui toimimispiirkond.

Generate Particles (loo osakesed)

Määrab, kui palju vedeliku osakesi luua (0=väljas, 1=tavaline, >1=rohkem). Et seda kasutada, peab pinnatükeldamise väärtus olema vähemalt 2.



Siin on näha osakeste efekti - vasakpoolne pilt on simuleeritud ilma ja parempoolne koos osakeste ning pinnatükeldusega.

Vedelik

Kõiki selle objekti osasid, mis asuvad seespool toimimispiirkonna piirdkasti, kasutatakse simulatsioonis vedelikuna. Kui lisad toimimispiirkonnale rohkem kui ühe vedeliku, siis hetkel ei peaks nad lõikuma. Samuti veendu, et pinnanormaalid osutaksid väljapoole. Erinevalt toimimispiirkonna objektidele, kasutatakse vedelikuobjektide puhul tegelikku võre geomeetria.

Volume init (mahu algseadistamise tüüp)

- Volume Määrab objekti sisemise osa vedelikuna. See töötab ainult suletud objektidega.
- Shell (kest) määrab vedelikuna ainult õhukese kihi võre kõikidel külgedel. See töötab ka lahtistel võredest.
- Both (mõlemad) kombineerib mahu ja kesta - sel juhul peaks võre olema samuti suletud. Vaata allpool asuvat pilti.



Erinevate mahu algseadistamise tüüpide näide: maht (Volume), kest (Shell) ja mõlemad (Both). Pane tähele, et kest on tavaliselt natuke suurem kui maht.

Initial velocity (algkiirus)

Vedeliku kiirus simulatsiooni alguses, meetrites sekundi kohta.



Pinnanormaalide suunal on suur vahe!

Blender kasutab pinnanormaalide suunda, et otsustada, mis on vedelikuobjekti "sees" ja mis on "väljas". Sa tahad, et kõik normaalid osutaksid *väljapoole* (vajuta muutmisrežiimis kiirklahvi Ctrl+N või vajuta Space ja vali Edit → Normals → Calculate Outside (muuda->normaalid->arvuta väljapoole)). Kui normaalid osutavad vales suunas, saad sa "hiiglasliku veeuputuse", sest Blender arvab, et objekti maht on väljaspool selle võret! See toimub sõltumata mahu tüübi (Volume init) tüübi seadetest.

Takistus

Seda objekti kasutatakse simulatsioonis takistusena. Vedelikobjekt ja takistus objekt ei tohiks hetkel üksteist läbistada. Nagu vedelikuobjektide puhul, kasutatakse ka takistuste puhul tegelikku võre geomeetria. Mahuga objektide puhul veendu, et takistuse normaalid oleksid arvutatud õigesti ja näitaksid õiges suunas (kasuta muutmisrežiimis võretööriistade (Mesh Tools) paneelil muutmiskontekstis Editing nuppu Flip Normal (pööra normaalid ümber) (F9)), eriti kui kasutad pööramisvahendiga tehtud mahutit. Kui võre ei ole animeeritud, on enne eeltöötlust hea mõte rakendada pinnatükeldajat (SubSurf).

Volume init (mahu algväärtustamise tüüp)

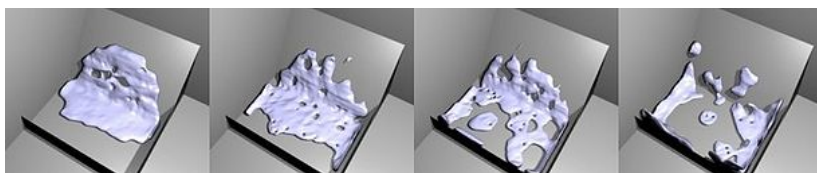
Sama mis ülalkirjeldatud vedelikuobjekti puhul.

Boundary type (piiri tüüp) (vaata pilti allpool)

Määrab takistuse pinna kleepuvuse, nimetatakse ka "pinna adhesiooniks". Reaalses maailmas sõltub pinna adhesioon vedelikust ja pinna teralisusest või hõõrdumise/adhesiooni/imevuse väärtustest.

- Noslip (mittelibisev) põhjustab vedeliku kleepumise takistuse külge (nullkiirus).
- Free (vaba libisemine) lubab liikumist piki takistuse pinda (ainult kiirus normaali suunas on null).
- Part (osaline libisemine) segab mõlemad tüübid, kus väärtus 0 on peamiselt mittelibisev ja 1 vabaltlibisev.

Pane tähele, et kui võre liigub, koheldakse seda automaatselt mittelibisevana.



Näide erinevate piiri tüüpide mõjust, kui tilk kukub kaldpinna. Vasakult paremale: mittelibisev, osaline libisemine 0.3, osaline libisemine 0.7 ja vaba libisemine.

Animated Mesh/Export (animeeritud võre/eksport)

Klõpsa seda nuppu, kui võre on animeeritud (näiteks moonutatud skeleti, võtmevormi või sõrestiku poolt). Pane tähele, et see on märgatavalt aeglasem ja ei ole vajalik, kui võre on animeeritud ainult positsiooni või pöörmise IPOdega (st ainult *objekti* teisendustega).

PartSlip Amount (osalise libisemise määr)

Eelpool kirjeldatud mittelibisemise ja vabaltlibisemise segamise hulk.



Blender 2.5 toetab nüüd liikuvaid takistusi!

Kui takistus liigub, kohtleb Blender seda automaatselt mittelibisevana (kleepuvana). Kui tahad, et vedelik pritsiks liikuvast objektist eemale, pane sinna, kus vedelik liikuva objektiga kokku puutuks, läbipaistev tasapind, mis on täpselt sama kujuga kui objekt, kuid veidike objektist ettepoole, vedeliku ja objekti vahele. Kui objekt tuhiseb mööda ja vedelik pritsib, siis pritsimine puutub tegelikult vastu läbipaistvat pinda ja libiseb eemale ning objekt jätkab oma teed

Impact Factor (mõjufaktor)

Vedeliku mahu suurenemise/vähendamise korrigeerimine liikuva objektiga kokkupõrkel. Kui objekt ei liigu, ei ole sellel seadel mingit efekti. Kui ta aga liigub ja vedelik temaga kokku põrkab, siis negatiivne väärtus vähendab vedeliku mahtu toimumispiirkonnas ja positiivne suurendab seda. Vahemik on -2.0 kuni 10.0.

Sissevool

See objekt lisab simulatsioonile vedeliku (umbes nagu veekraan).

Volume init (mahu algväärtustamise tüüp)

Sama mis ülalkirjeldatud vedelikuobjekti puhul.

Inflow velocity (sissevoolu kiirus)

Kiirus, millega objekti sees vedelikku luuakse.

Local Coords/Enable (kohalik koordinaadistik/lülita sisse)

Kasuta sissevoolul kohalikke koordinaate. See on kasulik, kui sissevoolu objekt liigub või pöördub, sest siis sissevoolu juga järgib/kopeerib seda liikumist. Kui see on välja lülitatud, siis sissevoolu asukoht ja suund ei muutu.

Animated Mesh/Export (animeeritud võre/eksporti)

Sama mis ülalkirjeldatud takistusobjekti puhul.

Väljavool

Vedelik, mis selle objekti piirkonda satub, kustutatakse (nagu äravool või must auk). Seda kasutatakse kombineerituna koos sissevooluga, et vältida toimumispiirkonna vedelikuga täitumist. Kui see on sisse lülitatud, näeb ta välja nagu tornaado (vesipüks) või tolmuimeja ja see osa, kust vesi kaob, järgib objekti, kui see ringi liigub.

Volume init (mahu algväärtustamise tüüp)

Sama mis ülalkirjeldatud vedelikuobjekti puhul.

Animated Mesh/Export (animeeritud võre/eksporti)

Sama mis ülalkirjeldatud takistusobjekti puhul.

Osake

Seda tüüpi kasutatakse simulatsiooni käigus loodud osakeste kuvamiseks. Praegu toetatakse ainult osakesi, mis ujuvad koos vedelikuga. Pane tähele, et sellel objektil võib olla suvaline kuju, asukoht või tüüp: kui vajutada osakeste nuppu, luuakse osakeste süsteem koos vedeliku simulatsiooni osakestega õigesse kohta. Kui algset objekti liigutada, võib osutada vajalikuks kustutada osakeste süsteem, välja lülitada vedelikusimulatsiooni osakesed ja siis need uuesti sisse lülitada. Teised osakeste jõud või seaded ei mõjuta vedelikusimulatsiooni osakesi.

Particle type (osakeste tüüp)

Drops (tilgad)

Vedeliku pinna pritsimisel tekivad tilgad, mis on laiali külvatud nagu madala pindpinevusega puhta vee puhul.

Floats (pritsmed)

Vedeliku pindpinevus on kõrgem ja vedelik raskem, nagu külm merevesi ja supp. Eraldunud osad on ebakorrapärasemad ja langevad pinnale tagasi kiiremini kui tilgad (Drops), nagu suure pindpinevuse puhul.

Tracer (jalgija)

Piisad järgivad seda kohta, kus vee pind on olnud, nagu udu, mis ripub eelmiste vedelike tasemetel kohal. Kasuta seda, et näha, kus vedeliku tase on olnud.

Size Influence (mõju suurusele)

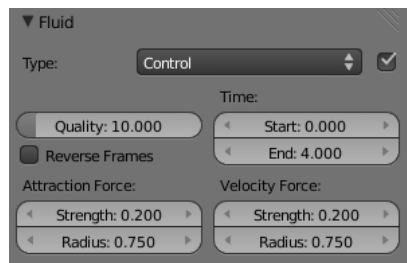
Osakesed võivad olla erineva suurusega, kui selle seade väärtus on 0, siis on nad kõik sama suurusega.

Alpha Influence (mõju katvusele)

Kui selle seade väärtus on >0, muudetakse osakeste katvuse (alfa) väärtust vastavalt nende suurusele.

Path (eeltöötuse kataloog)

Simulatsiooni tulemus, kust laaditakse osakesed. Siin peaks harilikult olema sama väärtus, mis vedeliku toimimispiirkonna objektile (näiteks kopeeri see kiirklahvidega CtrlC, CtrlV).

Juhtimine

Vedeliku juhtimise seaded.

Quality (kvaliteet)

Kõrgema kvaliteedi tulemusena luuakse juhtobjekti jaoks rohkem juhtosakesi.

Reverse Frames (tagurpidi kaadrid)

Juhtosakeste liikumine muudetakse vastupidiseks.

Time (aeg)

Saad määrata alguse ja lõpu aja, mille jooksul vedeliku juhtobjekt on aktiivne.

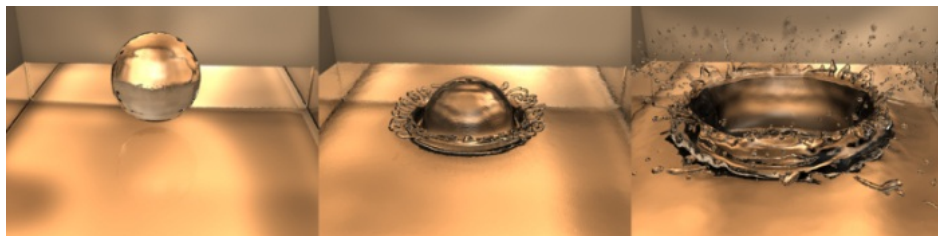
Attraction force (kõrgetõmbejõud)

Kõrgetõmbejõud määrab, millist jõudu vedeliku juhtobjekt emiteerib. Positiivne jõud tõmbab vedelikku ligi, negatiivne välidib.

Velocity force (kiiruse jõud)

Kui vedeliku juhtobjekt liigub, siis tema kiirus avaldab ka vedelikule jõudu.

Vaata täpsemat informatsiooni [nendest väljalaskemärkustest](#).



Veel üks näide kukkuvast tilgast animatsioonist, mis on simuleeritud Blenderi ja renderdatud Yafray abil.

Vedeliku omaduste muutuste animeerimine

Vedeliku toimimispiirkonna objekti jaoks on olemas uut tüüpi IPO-kõver, FluidSim (vedelikusimulatsioon). Erinevalt teistest animeeritavatest väärtustest Blenderis, ei saa FluidSim IPOle võtmekaadrit määrata ainult kiirklahvi abil, vaid sa pead IPO aknas väärtused käsitsi sisestama. Et võtmekaadrit määrata, pead sa IPO aknas valima omaduse, mida soovid animeerida, ja klõpsama Ctrl LMB, et seada võtmekaader IPO aknas soovitud kohta.

**Sisesta omadused**

Pane tähele, et ei pea olema täpne, kuhu sa klõpsad; soovitage pärast kontrollpunkti seadmist avada teisenduse omaduste paneeli Transform Properties (N), ümardada X-väärtus täisarvulise kaadri numbrini ja siis määrata Y-väärtus, nii nagu soovid.

Vedeliku toimimispiirkonnal on mitu kanalit, mis juhivad vedelikku ajas:

Fac-Visc (viskoossuse koefitsient)

Vedeliku viskoossuse koefitsiendi korrutav parameeter. See tuleb seada enne eeltöötust ja muudab aja jooksul vedeliku viskoossust, nii saad näiteks muuta vee õliks!

Fac-Tim (ajakoeffitsient)

Muudab simulatsiooni kiirust; nii nagu kiiruse seade videorektoris suurendab või vähendab video kiirust, suurendab või vähendab see kõver vedeliku liikumise kiirust kaadrite järjestuses. Kui Fac-Tim väärtus on väiksem või võrdne nulliga, siis aeg (ja vedelik) seisab paigal; vedelik tardub. Vahemikus 0.0 kuni 1.0 vahel voolab vedelik aeglasemalt ja paistab tihedam. 1.0 on normaalne vedeliku liikumine ja väärtused, mis on suuremad kui 1.0, panevad vedeliku voolama kiiremini ja võimalik, et paistma õhemana.

GravX/GravY/GravZ (gravitatsiooni X/Y/Z)

XYZ-vektor gravitatsiooni muutusteks ehk vedeliku enda inerts (kujutle kohvijoomist ringrajaauto roolis, espresso lonksamist kiirteel või taimede kastmist kosmosesüstikus). Tehes muudatusi nendes kõverates, paned sa vedeliku väliste jõudude toimele loksuma.

The Fluid (vedelik), Obstacle (takistus), Inflow (sissevool), Outflow (väljavool) ja Particle (osake) tüüpi objektid võivad kasutada järgmisi kanaleid:

VelX/VelY/VelZ (kiirus X/Y/Z)

Vedeliku purskumist alavoolikust saab simuleeruda nende kõverate abil, imiteerimaks muutusi surves ja/või suunas. Seda saab näiteks kasutada ka tuule mõju simuleerimiseks veejoale.

Active (aktiivne)

Kui aktiivsus läheb väärtuselt 0.0 millekski suuremaks kui 0 (nagu näiteks 0.1 ja 1.0 vahel), hakkab vastav objekt (määratud kui Inflow (sissevool) või Outflow (väljavool) jne) jälle toimima. Langedes alla 0.0-ni ja tõustes siis mingil hetkel uuesti üles, taastub tema mõju ja sellega kaasnev vedeliku simulatsioon. Kasuta seda tilkumise või mõne muu katkendliku sissevoolu korral. See aktiivne olek töötab ka objektidel, mis on määratud kui Outflow (väljavool) ja Obstacle (takistus) - nii saad sa (näiteks) simuleerida äravoolule korgi ette panemist.

Samuti saad määrata juhtobjektide Control jõu seadeid:

AttrForceStr, AttrForceRa

Need kõverad juhivad külgetõmbejõu seadete väärtusi.

VelForceStr, VelForceRa

Need kõverad juhivad kiiruse jõu seadete väärtusi.

Tehnilised detailid



"My cup runneth over", loodud Blenderi ja Yafray abil.

Vedelikusimulatsioon võtab palju aega - mida paremini sa selle tööd mõistad, seda lihtsam on sul hinnata, kui realistlik tulemus saab olema. Algoritm, mida Blender vedeliku simulatsioonis kasutab, on *Boltzmanni meetod võrel* (LBM); teised vedeliku algoritmid on *Navier-Stokesi* (NS) lahendajad ja *Silutud osakeste hüdrodünaamika* (SPH) meetodid. LBM on kuskil nende kahe vahepeal.

Üldiselt on praegusel arvutitel väga raske simuleerida isegi 1-meetrist veetünni. Et simuleerida lainet murdmas läbi linna, on sul ilmselt vaja üht kõige kallimatest superarvutitest, mis saada on, ja ka see ei pruugi ikkagi töötada õigesti, ükskõik millist neist kolmest algoritmist sa kasutad. Seetõttu pead sa sellise efekti saavutamiseks, mida sa tegelikult soovid, võtma abiks strateegia, mis sarnaneb "analoogfilmides" palju aastaid kasutatud meetodiga: "võltsi seda!"

Hea vedelikusimulatsioon on *väga tähtis*, kuid mitte *ainuke* osa, et saavutada rahuldav pilt. Lase Blenderil teha arvutamise must töö vedeliku baassimulatsiooni arvutamisel, seejärel muuda see tõetruuks, lisades valitud detaile, mis vastavad vaataja ootusele "reaalsele veekeerise osas, mille sa oled loonud".

Näiteks saad sa teeselda, et sul on linnas hiidlaine, ehitades *väiksema* mudeli, modelleerides *väikese* laine linna mudelisse väga

kõrge lahutusega ja loota, et keegi ei märka erinevust 100 m ja 1 m laine vahel (ei märkagi). Lisa lainefrondile tekstuur müra ja pilvedega, mis mõjutavad värvust. Lisa rohkesti suitsu (udu) emiteerijaid erinevatele pindadele, mida laine tabab, ajastades need emiteerima kokkupõrke hetkel kokkupõrke ja pinna suunas. Animeeri autod ja prügi (ja uppuvad inimesed...) lainefrondile hulpima ja ujuma, kasutades eeltöödeldud võret. Kasuta laine harjal rida udu emiteerijaid, et simuleerida veeudu, mis laine harjalt õhku paiskub. Mõtle täpselt järele, kuhu sa tahad paigutada kaamera, kas kasutada suumobjektiivi või lainurka ja nii edasi (kas vaataja "vaatab vaestele näitlejatele ülalt alla" või "upub koos nendega"?). Sedasorti tähelepanu muudele detailidele peale vedelikusimulatsiooni enda teebki võttest selle, mis ta olema peab.

Blenderi LBM-i lahendajale teevad arvutamise raskemaks järgmised asjad:

- Suured toimimispiirkonnad.
- Pikad kestused.
- Madalad viskoossused.
- Suured kiirused.

Vee viskoossus on niigi madal ja eriti väikse lahutuse korral ei tabata vee turbulentsi päris õigesti. Kui vaatad lähemalt, siis enamik vedelikusimulatsioone ei paista praegu väga reaalse vee moodi. Üldiselt, ära tugine liiga palju füüsilistele seadetele (nagu toimimispiirkonna füüsiline suurus või animatsiooni pikkus sekundites). Parema ürita saavutada üldine meeleolu madala lahutusega ja siis suurenda seda nii palju kui võimalik või soovitud.

Vihjed

- Ära ehmata, sa saad pärast simulatsiooni terve portsu võre (.bobj.gz) faile. Üks komplekt eelvaate ja teine lõpliku renderduse jaoks. Igas komplektis on üks .gz fail iga animatsiooni kaadri jaoks. Iga fail sisaldab simulatsiooni tulemust - seega sa vajad neid.
- Praegusel hetkel ei kustutata neid faile automaatselt, seega on hea mõte luua simulatsiooni tulemuste säilitamiseks spetsiaalne kataloog. Vedelikusimulatsiooni tegemine on nagu nupu ANIM klõpsamine - praegu pead sa vedeliku pinna võred ise kuhugi kataloogi panema. Kui soovid simulatsiooni kasutamise lõpetada, võid lihtsalt kustutada kõik *fluid*.bobj.gz failid.
- Enne kõrge lahutusega simulatsiooni jooksutamist, mis võib võtta tunde, kontrolli üldist ajastust, jooksutades esmalt madala resolutsiooniga.
- Vedelikuobjektid peavad olema täielikult toimimispiirkonna objekti piirdkasti sees. Kui ei ole, ei pruugi eeltöötlus õigesti toimida. Vedelikobjektid ja takistused võivad olla keeruka geomeetriaga võred. Väga õhukesed objektid ei pruugi simulatsioonis näha olla, kui valitud lahutus on liiga jäme (lahutuse suurendamine võib selle probleemi lahendada).
- Pane tähele, et selliseid vedelikusimulatsiooni parameetreid nagu sissevoolu kiirus saab animeerida Fluidsim IPOdega (vaata ülaltpoolt).
- Ära ürita teha keerukat stseeni ühe korraga. Blenderil on mitmekülgne komposiitor, mida saad kasutada mitme animatsiooni kombineerimisel.

Näiteks selleks, et teha animatsioon, mis näitab kahte erinevat vedeliku voolu, hoides samal ajal toimimispiirkonda väiksena, renderda üks .avi-fail, kasutades ühte voolu. Siis liiguta toimimispiirkonda ja renderda teine .avi-fail teise vooluga, kasutades alfakanalit (eraldi mustvalge .avi?). Siis komposiidi mõlemad .avi-failid, kasutades komposiitori liitmiskompositsiooni. Kolmas .avi on harilikult suitsu ja udu jaoks ning pannakse ka kõige peale. Lisa veel udule vihmakiht ja pritsmed ja sul on päris torm möllamas! Ja siis välgulöögid, ringilendlev praht, kõik eraldi animatsioonid, komposiitid kokku tõeliselt suurepäraseks tulemuseks.

- Kui sul on probleeme või miski ei tööta nii nagu vaja - anna mulle teada: saada .blend-fail ja probleemi kirjeldus `nils at thurey dot de`. Vaata selle viki lehekülgi ja [blenderartists-foorumit](#) enne maili saatmist!

Piirangud ja lahendused

- Ühe Blenderi faili kohta üks toimimispiirkond (versioon 2.42), aga sul võib olla mitu vedelikuobjekti.

Lahendus: tõsta eelvaateks toimimispiirkonda ringi, et ta hõlmaks iga vedelikuvoolu osa, lõpuks suurenda toimimispiirkonda nii, et ta sisaldaks kõiki vedelikuobjekte (kuid arvutusaeg tuleb pikem). Õigupoolest on see eelis, sest varieerides toimimispiirkonna suurust ja asukohta, saad sul valida, kui palju arvutusaega kasutatakse.

- Kui süsteem paistab valesti minevat, tee kindlaks, et kõik normaaliid on õiged (mine muutmisrežiimi, vali kõik ja arvuta normaaliid uuesti).
- Hetkel on probleeme nullgravitatsiooniga simulatsiooniga - vali lihtsalt väga väike gravitatsiooni suurus, et seda parandada.
- Kui objekt on määratud kui maht (Volume), peab see olema suletud ja tal peab olema sisemine külg (tasapind ei toimi). Et kasutada tasapindu, lülita tüübiks kest (Shell) või suru tasapind välja.
- Blender hangub pärast nupu BAKE klõpsamist. Vajutades Esc, hakkab ta mõne aja pärast uuesti tööle - see võib juhtuda, kui lahutus on liiga kõrge ja mälu vahetatakse kõvakettale, tehes kõik väga aeglaseks. Sel juhul peaks aitama lahutuse vähendamine.
- Pärast nupu BAKE klõpsamist jookseb Blender kokku - see võib juhtuda, kui lahutus on tõeliselt kõrge ja hõivatud on rohkem kui 2 GB mälu, põhjustades Blenderi kokkujooksmise. Vähenda lahutust. Pea meeles, et mitmetel operatsioonisüsteemidel on piiratud see, kui palju mälu ühele "protsessile", nagu näiteks Blenderile eraldatakse, isegi kui "masinal" on mälu sellest rohkem. Nõme...
- Võred peaksid olema suletud - seega, kui näiteks vedelikuobjekti mõned osad ei toimi simulatsioonis vedelikuna, siis kontrolli, kas kõik ühendatud tippude grupid on suletud võred. Kahjuks ei ole Suzanne'i (ahvi) võre Blenderis suletud võre (silmad on eraldi).

- Kui vedeliku simulatsioon lõpetab veateatega (näiteks "init has failed"), tee kindlaks, et toimimispiirkonna objekti seaded on õiged, näiteks taastades algseaded.
- Et importida üksikut vedeliku võret, võid kasutada seda skripti: [.obj-Import-Script](#).
- Võib juhtuda, et sa ei saa eeltöödelda vedelikku, mis võtab mälu rohkem kui 1 GB, isegi mitte siis, kui Blender on kompileeritud LargeAddressAware'i seadetega - see võib olla praeguse vedelikumootori piirang.
- Pane tähele, et esimene kaader võib võtta mälu ainult mõnisada MB, kuid hilisemad lähevad juba üle ühe GB, mis võib olla põhjuseks, miks eeltöötlus ebaõnnestub. Kui nii, siis proovi eeltöödelda üks kaader keskelt või lõpust täislahutusega, et näha, kas see töötab.
- Mälu kasutus kahekordistub, kui sead pinnatükeldamise 1st 2le.
- "Osakeste genereerimine" suurendab samuti mälu kasutamist, kuna need suurendavad pinda ja keerukust. Tavalised vedelikusimulatsiooni osakesed võtavad ilmselt vähem mälu.

TEHA: kontrolli neid linke...(Et nad oleks sobivad Blender 2.5.ga

Vaata ka

- [Õppetükk 1: Väga algeline sissejuhatus](#)
- [Õppetükk 2: Järgmine samm](#)
- [Õppetüki 1&2 kasutajaliidese muudatused uute versioonide jaoks](#)
- [Veel üks BSoD vedeliku õppetükk](#)
- [Arendusdokumentatsioon \(lahendus, vajalikud komponendid jms\)](#)

Väline dokumentatsioon

- [An Introduction to Fluid Simulations in Blender \(video\) \(Blendernation link\)](#) - Õpi, kuidas luua Blenderis vedelikusimulatsiooni koos takistusega.
- [Fluid Simulator Tutorial \(video\) \(Blendernation link\)](#) - Väga lihtsasti arusaadav algajate videoõppematerjal vedelikusimulatsioonist. Käsitleb ka mõningaid enamlevinud vigu.
- [Guide on Blender Fluid Simulator's Parameters \(Blendernation link\)](#)

Tänu sõnad

Vedelikusimulaatori integreerimine tehti Google Summer-of-Code projektina. Rohkem infot lahendaja kohta leiab www.ntoken.com. Need animatsioonid on tehtud enne lahendaja integreerimist Blenderisse: [Adaptive Grids](#), [Interactive Animations](#). Tänu Chris Wantile Blender-SoC projektide organiseerimise eest ja Jonathan Merritile selle mentoriks olemise eest! Ja muidugi tänu Google'ile kogu asja alustamise eest... SoC edenemine ja uuendused: [SoC-Blenderfluid-Blog at PlanetSoC](#).

Lahendaja ise on arendatud Nils Thuerey poolt ja järgnevate inimeste abiga ning järelvalve all: U. Ruede, T. Pohl, C. Koerner, M. Thies, M. Oechsner ja T. Hofmann, Arvutiteaduse osakonnas 10 (System Simulation, LSS) Erlangenis Saksamaal.

<http://www10.informatik.uni-erlangen.de/~sinihue/img/lsslogo.png>

<http://www10.informatik.uni-erlangen.de/~sinihue/img/unierlangenlogo.png>

Ookeanisimulatsioon (*Ocean Simulation*)

Blenderi ookeanisimulatsioon on töötleja, mis võimaldab moodustada muutuva ookeanipinna koos vajalike tekstuuridega ja kasutada seda renderdamisel. See on porditud avatud lähtekoodiga programmist *Houdini Ocean Toolkit* ja selle eesmärk on simuleerida ookeanilaineid ja merevahtu.

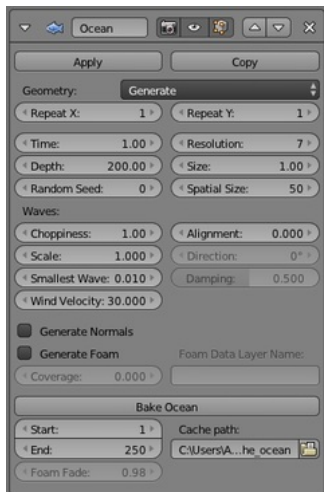
Simulatsiooni tööpõhimõte

Simulaator genereerib sisemiselt FFT (*Fast Fourier Transformation* — kiire Fourier' teisendus) abil 2D-andmete ruudustikud (kaardid), mis sarnanevad 2D-tekstuuridele. Genereeritakse kolme tüüpi andmed: pinna ümberpaigutus, pinnanormaalid ja lisaandmed, millest arvutatakse laineharjade löiked (st vaht). Pärast simuleerimist kasutatakse neid kaarte ookeani pinna ümberpaigutamiseks 3D-ruumis. Neid on võimalik kasutada ka varjutamisel (*shading*) ookeanitekstuurina. Sisemiselt on simulatsioonimootor mitmelõimeline (*multithreaded*) ja kasutab OpenMP teegi abil mitut protsessorituumat.

Ocean Modifier - ookeanitöötletaja

Mode: Object Mode (Objektirežiim)

Kirjeldus



Ookeanitöötletaja paneel

Ocean Modifier on peamine koht Blenderis, kus vastav simulatsioon toimub. Töötletaja haldab simulatsiooniandmeid ja rakendab neid deformeeritud ookeanipinna võre loomisel. Ookeanitöötletaja võib merevahu visualiseerimiseks lisada võrele ka tippude värvi kanali.

Geomeetria valikud

Geometry (geomeetria)

Ookeanitöötletaja võib võre geomeetriat muuta kahel viisil:

- Genereerida mosaiikse pinnavõre, mis vastab täpselt simulatsiooniandmete lahutusele (generate).

Sellisel juhul kirjutatakse algne võre täielikult üle ookeanipinna võrgustikuga. Võrele lisatakse ka UV-kanal, mis teisendab [0,0,1.0] UV-koordinaadid simulatsioonivõre koordinaatideks.


- Nihutada olemasoleva, suvalise topoloogiaga võre tippe (*displace*).

Repeat X, Repeat Y (X kordus, Y kordus)

Määrab võre pinna genereerimisel, mitu korda ruudustikku X ja Y suundades korratakse. Nende korratud võreosade UV-koordinaadid jätkuvad väljaspool [0,1] UV-ruumi.

Simulaatori valikud

Time (aeg)

Ajaketk, mida ookeanisimulatsioon parajasti esitab. Et teha animeeritud ookeani, tuleb määrata võtmekaadrid (RMB ) ja animeerida aja väärtust. Selle muutumise kiirus määrab, kui kiiresti lained liiguvad.

Resolution (lahutus)

Põhiline seadistus, mis mõjutab simulatsioonimootori kiiruse ja kvaliteedi suhet. Sellega määratakse sisemiste 2D-andmeruudustike suurus. Sisemiste ruudustike mõõtmed on 2 astmes lahutus: st lahutus 8 loob 256×256 andmeruudustikud. Mida suurem on lahutus, seda kauem arvutused aega võtavad, kuid seda rohkem detaile simulatsioon genereerib.

Märkus: kui kasutada geomeetria valikut *Generate* (genereeri), määrab see seadistus ka genereeritud võre lahutuse (mis on võrdne sisemise andmeruudustiku lahutusega).

Spatial Size (ruumiline ulatus)

Simuleeritud ookeanipinna ulatus meetrites. See määrab ka genereeritud võre või nihutatud ala suuruse Blenderi ühikutes. Muidugi on võimalik ookeanitöötletajaga võret objektirežiimis skaleerida, et stseeni näivat suurust muuta.

Depth (sügavus)

Ookeanipõhja püsiv sügavus simuleeritud ala all.

Lainete valikud

Choppiness (ebaühtlus)

Laineharjade ebaühtlus või teravus. Kui ebaühtlus on 0, liigub ookeanipind ainult üles ja alla piki Z-telge. Suurema ebaühtluse puhul nihutatakse laineid ka külgsuunas (X- ja Y-telje suunas), nii et moodustuvad teravad laineharjad.

Scale (suurus)

Lainete suuruse (amplituudi) üldine seadistus. See on keskmine lainete kõrgus või sügavus võrreldes merepinnaga. Erinevalt lihtsalt ookeani objekti skaleerimisest Z-telje suunas, mõjutab see seadistus ka teisi simulatsiooni aspekte, nagu lainete külgsuunaline nihutamine, vahuharjade moodustamine ja pinnanormaalide suund.

Alignment (joondus)

Lainete kuju suunatus tuule mõjul. Kui joonduse väärtus on 0, on lained juhuslikult ja igas suunas ühtlaselt orienteeritud. Suurema väärtuse korral puhub tuul tugevamalt ja ühtlasemalt samast suunast ning lainekujud on kokku pressitud ja orienteeritud vastavasse suunda.

Direction (suund)

Kui kasutatakse joondust, määrab lainete (tuule) suuna (kraadides).

Damping (sumbumine)

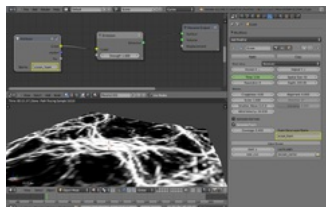
Kui kasutatakse joondust, määrab, kui palju omavahel peegeldunud lained sumbuvad. See muudab lisaks lainete kujule ka nende liikumise suunatuks. Kui sumbumine on 0.0, peegelduvad lained üksteiselt kõigis suundades. Kui see on 1.0, sumbuvad peegeldunud lained täielikult, jättes alles ainult tuule mõjul ühes suunas liikuvad lained.

Smallest Wave (väikseim laine)

Genereeritud laine minimaalne suurus. Käitub sarnaselt siluvale filtrile, eemaldades suure sagedusega väikese lainesäbru.

Wind Velocity (tuule kiirus)

Tuule kiirus (meetrites sekundis). Väikese tuulekiiruse korral moodustuvad ainult väiksemad pinnalained.

Simulatsiooniandmete loomise valikud

Vahu tipuvärvide kasutamine koos määratud andmekihiga.

Vaikimisi genereerib simulaator ainult pinna nihke andmed, sest see nõuab kõige vähem arvutusi ja annab kiireima tagasiside. Renderdamise jaoks võib genereerida lisaandmeid.

Generate Normals (genereeri normaalid)

Loob lisaks pinna normaalikaardi. Selle võib lisada faktuurina ookeani tekstuurile, kui kasutada normaalikanalit. Eeltöötluse käigus on võimalik genereerida seeria normaalikaarte (*normal maps*).

Generate Foam (genereeri vaht)

Loob lisaks lainevahu kaardi. Seda saab kasutada ookeani tekstuuris (näiteks maskina). Eeltöötluse (*baking*) käigus on võimalik genereerida seeria vahukaarte.

Coverage (katvus)

Määrab, kui palju lainest on vahuga kaetud. Negatiivsed väärtused vähendavad vahu hulka (jättes ainult vahused laineharjad), positiivsed suurendavad vahu hulka.

Foam Data Layer Name (vahu andmekihi nimi)

Tipuandmete kihi nimi, mida ookeanitöötaja kasutab vahukaartide ja tipuvärvide salvestamiseks. See on vajalik renderdajas vahuandmetele ligipääsemiseks.

Eeltöötlus (*baking*)

Selle asemel, et ookeanipinda reaajas simuleerida, võib andmed arvutada kettale. Kui simulatsioon on eeltöödeldud, siis simulatsioonimootorit ei kasutata ja kõik töötaja ja teksturi andmed loetakse eelgenereeritud failidest.

Eeltöötlamine võib olla kasulik mitmel põhjusel:

- Salvestatud andmete kasutamine on kiirem kui nende genereerimine
- Ookeani andmeid on võimalik kasutada välistes renderdusprogrammides
- Võimaldab kasutada täiuslikumaid vahukaarte

Andmefailid

Andmed salvestatakse kettale rea OpenEXR pildifailidena, üks fail iga nihke, normaalikaardi ja vahukaardi kohta (kui neid kasutatakse). Kui eeltöödeldud andmeseeria kaader on kord kettalt loetud, puhverdatakse seda edaspidi mälus. Tänu sellele on kord loetud kaadrite edaspidine kasutamine kiire (puudub kettalt lugemise viivitus).

Kuna need failid on standardsed OpenEXR-pildid, saab neid avada ja renderdada iga programmiga, mis toetab OpenEXR-failiformaati.

Vahu eeltöötlus

Eeltöötlus võimaldab kasutada täiuslikumat vahtu. Reaalajas simuleerimisel kasutab ookeanisimulaator ainult käesoleva kaadri andmeid. Vahukaardi puhul tähendab see käesoleva kaadri laineharjade tippe. Tegelikuses, kuna vaht moodustub lainete lõikumisel, jääb ta mõneks ajaks laine pinnale, kuni aegamööda kustub. Eeltöötuse käigus on võimalik seda olukorda lähendada, jättes vahu eelnevatest kaadritest laine pinnale.

[\[video link\]](#)

Eeltöötuse valikud

Start, End (algus, lõpp)

Esimene ja viimane simulatsioonikaader. Kui püüda kasutada kaadreid väljaspool eeltöödeldud lõiku, korraldatakse esimest või viimast kaadrit.

Cache Path (puhvri asukoht)

Kaust, kuhu salvestada eeltöödeldud EXR-failid. Failinimed on kujul disp_####.exr (nihe), normal_####.exr (normaalikaart) ja foam_####.exr (vahukaart), kus #### on neljakohaline kaadri number. Kui puhvrikausta ei ole olemas, see luuakse.

[\[video link\]](#)

Simuleeritud ja pildilaotustele eelarvutatud Blenderis, renderdatud 3Delight'is.

Ajalugu

Algse simulaatori kirjutas Drew Whitehouse programmi [Houdini Ocean Toolkit](#) jaoks. Hamed Zagbaghi portis simulaatori C-keelde ja kohandas Blender 2.4 koodi jaoks. Seda toetas ProMotion Studios/[Red Cartel](#) lühifilmi "Lighthouse" ("Majakas") tegemiseks. Blender 2.5 versioonile lisatud täiendusi ja parandusi on toetanud '[Save the Ocean Sim](#)' projekt.

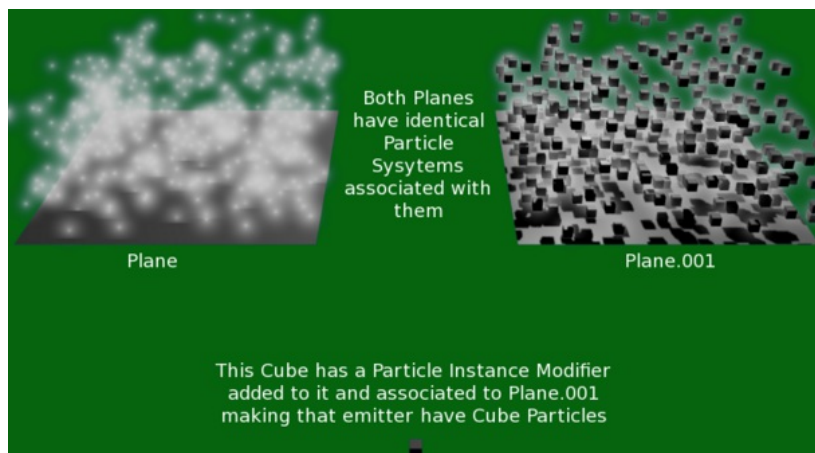
Particle Instance Modifier - osakesetöötleva

Mode: Kõik režiimid

Panel: Modifiers (Töötlevad) (Muutmiskontekst (Editing), F9)

Kirjeldus

Kui objektile on rakendatud Particle Instance Modifier, muudab see töötleva osakeste süsteemi osakesed objekti kujulisteks. See tähendab, et selle töötleva kasutamiseks peab stseenis olema mõni teine objekt koos osakeste süsteemiga, vastasel korral ei ole osakese töötleva mingit nähtavat tulemust.



Vasakpoolses osakeste süsteemis ei ole Particle Instance Modifier-objekti määratud. Parempoolses on osakeseks määratud kuup, lisades kuubile osakesetöötleva (Particle Instance Modifier).

Ülevaade

Allpool on lühike seletus terminitele, mida kasutatakse osakeste ja osakese töötleva (Particle Instance Modifier) kirjelduses:

- Osakeste süsteem (*Particle System*) – objekt (võre) millele on antud võime genereerida/väljastada osakesi.
- Tavaline osake (*Normal Particle*) – osake, mis ei ole teise osakese järeltulija.
- Alamosake (*Children/child particle*) – osake, mis on genereeritud ja paigutatud teise juba olemasoleva normaalse osakese järgi. Alamosakeste arvutamine on üldjuhul palju kiirem.
- Sündimata osake (*Unborn Particle*) – osake, mis ei ole veel nähtav/emiteeritud, sest tema väljastamise aeg ei ole veel käes. Osake võib olla sündimata olekus, kui animatsioon ei ole veel jõudnud tema tekkekaadri.
- Elus osake (*Alive Particle*) – osake, mis on nähtav/emiteeritud ja mis ei ole veel jõudnud surmaolekuni. Osake võib olla "elus", kui tema sünnist on möödunud vähem kaadreid, kui on tema eluiga.
- Surnud osake (*Dead Particle*) – osake, mida on näidatud/emiteeritud ja mille eluiga on läbi, st ta on jõudnud staatilisse lõppele.

Valikud

Kuna Particle Instance Modifier ja temaga seotud teiste objektide osakeste süsteemid mõjutavad üksteist vastastikku, võivad mõned osakesetöötleva efektid näha välja oluliselt erinevad, sõltuvalt selle osakeste süsteemi seadetest, milles vastavat osakest kasutatakse. Seda tasub silmas pidada, kui osakesetöötleva seadistused ei näi andvat soovitud tulemusi. Võimalik, et vajaliku efekti saavutamiseks on vaja muuta hoopis osakeste süsteemi, mitte osakesetöötleva seadistusi.

Object (objekt)

Väli Object (objekt) seob selle osakesetöötleva (Particle Instance Modifier) teise objektiga (tavaliselt sellise objektiga, millel on osakeste süsteem...). Seega, kui vastav objekt väljastab osakesi, on nende osakeste kuju määratud selle võre poolt, millele käesolev osakesetöötleva on rakendatud.

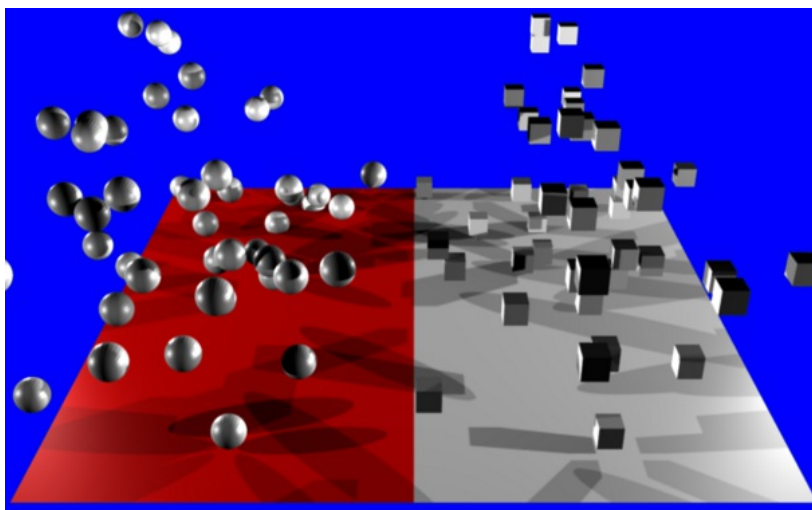
Näiteks kui töötleva ParticleInstance on rakendatud kerale ja selle töötleva objektiväljale on sisestatud teise osakesi emiteeriva objekti nimi, on tolles süsteemis emiteeritud osakesed kerakujulised.

Kuigi enamasti sisestatakse väljale Object (objekt) sellise objekti nimi, mis emiteerib osakesi, ei ole see kohustuslik. Blender ei kontrolli, et vastava objektile oleks osakeste süsteem rakendatud ja nii võib sellele väljale sisestada suvalise objekti nime.

Particle System (osakeste süsteem)

Väljal Particle System (osakeste süsteem) saab määrata, mitmendale teise objektiga seotud osakeste süsteemile osakesetöötlevat rakendada (juhul kui tolle objektiga on seotud rohkem kui üks osakeste süsteem). Välja Particle System (osakeste süsteem) väärtus võib olla vahemikus **1** kuni **10**. Siia saab sisestada suvalise numbriga lubatud vahemikus - Blender ei kontrolli, kas vastava numbriga osakeste süsteem teisel objektil ka tegelikult olemas on. Kui valida osakeste süsteemi number, mida olemas ei ole, on kõik teise objekti emiteeritud osakesed oma normaalse kujuga. Blender ei anna ka hoiatust, et vastava numbriga osakesesüsteemi ei eksisteeri.

Allpool toodud näites on tasand kahe eri piirkonnaga (üks punane ja teine valge), millest kummalegi on lisatud oma osakeste süsteem. Vasakpoolne kasutab kerale rakendatud osakesetöötlevat (Particle Instance Modifier) (osakesed on kerakujulised) ja parempoolne kuubile rakendatud osakesetöötlevat (osakesed on kuubikujulised).



Renderdus lihtsast tasandist, millel on kaks tipugruppi ja kummagi grupiga seotud iseseisev osakeste süsteem. Mõlemad osakeste süsteemid on seotud erinevate osakesetöötajatega. Antud näites on töötajaga töödeldud võred kera ja kuup.

[.blend-näitefail](#)

Loomine

Normal (tavaline)

Kui nupp Normal (tavaline) on aktiivne, joonistab ParticleInstance töötaja baasobjekti koopias iga kord, kui vastav osakeste süsteem väljastab tavalise osakese. Seega, kui käesolev osakesetöötaja on rakendatud kerale, on kõik emiteeritud normaalsed osakesed kerad.

Children (alamad)

Kui nupp Children (alamad) on aktiivne, joonistab osakesetöötaja baasobjekti koopias iga kord, kui vastav osakeste süsteem väljastab/kasutab alamosakest. Seega, kui käesolev töötaja ParticleInstance on rakendatud kerale, on kõik emiteeritud alamosakesed kerad.

Size (suurus)

Skaleerib osakest (koopiat) vastavalt seadistatule. Kui see on välja lülitatud, on kõik genereeritud koopiad sama suured kui originaal.

Esitus

Unborn (sündimata)

Kui nupp Unborn (sündimata) on aktiivne, joonistab ParticleInstance töötaja baasobjekti koopias iga kord, kui vastav osakeste süsteem väljastab/kasutab sündimata osakest. Seega, kui käesolev osakesetöötaja on rakendatud kerale, on kõik sündimata osakesed, kui nad näha on, kerad.

Alive (elus)

Kui nupp Alive (elus) on aktiivne, joonistab osakesetöötaja baasobjekti koopias iga kord, kui vastav osakeste süsteem väljastab/kasutab elusat osakest. Seega, kui käesolev osakesetöötaja on rakendatud kerale, on kõik nähtavad elusad osakesed kerad.

Dead (surnud)

Kui nupp Dead (surnud) on aktiivne, joonistab osakesetöötaja baasobjekti koopias iga kord, kui vastavas osakeste süsteemis esineb surnud osake. Seega, kui käesolev ParticleInstance töötaja on rakendatud kerale, on kõik nähtavad surnud osakesed kerad.

Trajektoori kasutamine

Create Along Paths (loo piki trajektoori)

Selle valikuga püüab osakesetöötaja muuta iga loodud osakese võret nii, et selle kuju sobituks vastava osakeste süsteemi liikumistrajektoori/karvadega. Alltoodud ekraanipildil on näha üksiku osakese paigutamine osakeste süsteemi kontrollpunkti **1** kuni **4** läbivale trajektoorige. Jõudes punkti **4**, osake sureb ja teekond lõppeb.

X,Y,X Rotation Axis (Pöörlemistelg X,Y,Z)

Määrab, ümber millise telje osake pöörleb.

Keep Shape (säilita kuju)

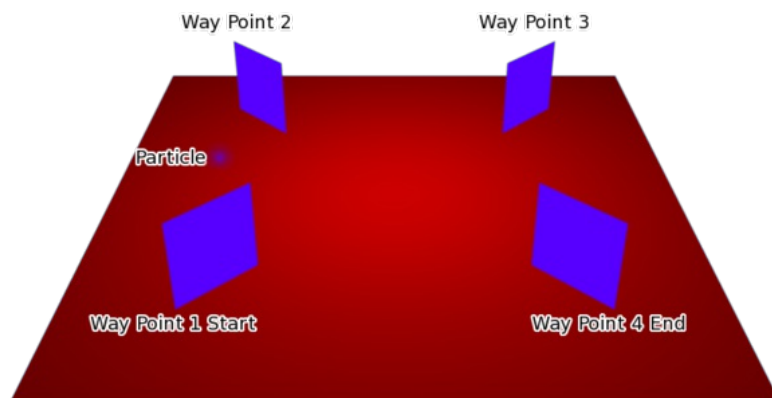
Kui see on sisse lülitatud, ei muuda osake oma kuju, vaid ainult pöörab ennast ümber keskpunkti läbiva telje trajektoori suunas.

Position (asukoht)

Määrab, kui suure osa trajektoorist (väljavenitatud) osake täidab. Selle väärtuse animeerimine ajas loob efekti osakese pikenemisest.

Random (juhuslik)

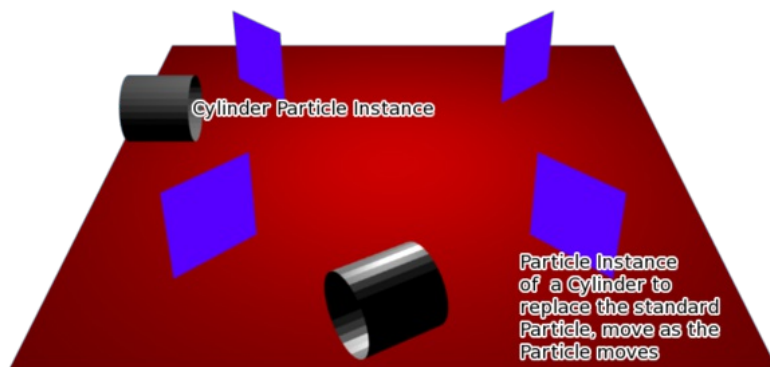
Skaleerib iga loodud osakese suurust (*Position*) juhusliku väärtusega.



Üksiku osakese paigutamine kontrollpunkte läbivale trajektorile.

[.blend-näitefail](#)

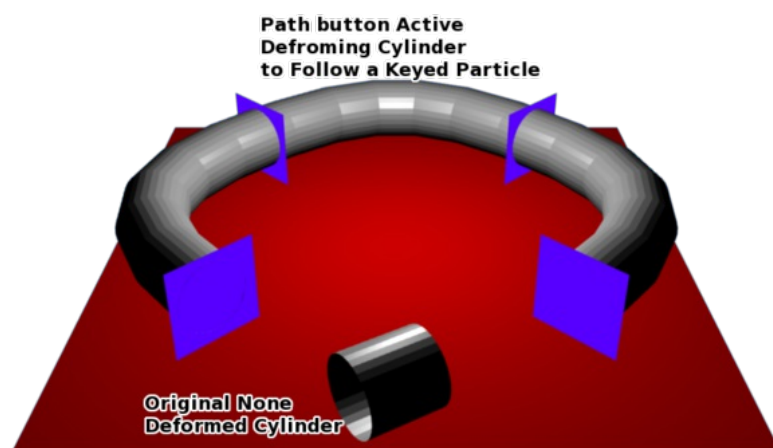
Kui osakesetöötaja on rakendatud silindrile ja seejärel liidetud kontrollpunktidega osakeste süsteemiga, määratakse silindri asukoht vastava osakese koordinaatidega. Seega järgib antud võre osakese asukohta. Osakest järgides ei muuda silinder ühtegi teist omadust - muutub ainult asukoht, mitte kuju ega pööre. Vaata alltoodud ekraanipilti:



Osakesetöötajaga määratud võre paigutamine kontrollpunkte läbivale trajektorile.

[.blend-näitefail](#)

Mõlemal ülaltoodud näitel on osakesetöötaja (ParticleInstance) trajektorinupp (Path) välja lülitatud. Kuidas käitub osake, kui nupp Path on aktiveeritud, on näha alltoodud ekraanipildil:



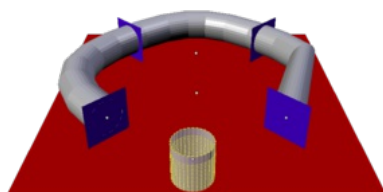
Osakesetöötlejaga määratud osakese paigutamine kontrollpunkte läbivale trajektoorige koos võre kuju sobitamisega vastavale trajektooriosale.

[.blend-näitefail](#)

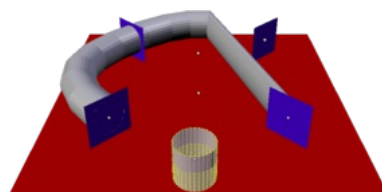
Nüüd silinder mitte lihtsalt ei järgi osakese trajektoori kuju ja asendit muutmata, vaid püüab sobitada oma võre kuju liikumistrajektooriga.

See, kui hästi kuju on võimalik sobitada, sõltub vastava objekti võre geomeetria tihedusest. Antud silindril on piki telge palju silmuslõikeid (*loop cut*), seega on teda võimalik vastavalt trajektoorige painutada.

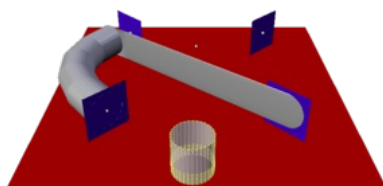
Sellel näitel on sama stseen mis eelmisel, kuid silindri silmuslõigete arvu on vähendatud. Silindri deformatsioon piki trajektoori on hästi nähtav.



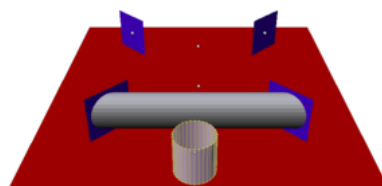
Enamik silindri silmuslõikeid on alles ja tema deformatsioon piki trajektoori on ühtlane, v.a lõpuosa.



Osad silmuslõiked on eemaldatud ja deformatsioon piki trajektoori ei ole enam sujuv.



Nüüd on deformatsioon piki trajektoori väga ebaühtlane.



Siin ei ole silindril enam ühtegi silmuslõiget (tippu), mille abil teda deformeerida, ja silinder liigub otse viimasesse kontrollpunkti **4**.

Kui kõik silmuslõiked on eemaldatud, jättes alles ainult alumise ja ülemise põhja tipud, ei võimalda silindri geomeetria teda enam painutada. Sel juhul ei järgi ta osakese trajektoori, vaid liigub otse kontrollpunktist **1** viimasesse kontrollpunkti **4**.

Osakesetöötaja (ParticleInstance) trajektoornupp (Path) töötab nii kiudude kui ka stoppkraadritega animeeritud osakeste puhul. Kiudude korral järgib osakesetöötaja kiudude pikkust ja kuju.

Allpool on näha ekraanipilt nupu Path (trajektoori) mõjust kiududele:



Kiud, millega on seotud osakesetöötaja, mis deformeerib silindrit piki kiu trajektoori.

[.blend-näitefail](#)

Märkus

Kiud surevad samal hetkel, kui nad luuakse, seega peab selleks, et nupp Path (trajektoor) neid mõjutaks, nupp Dead (surnud) olema samuti sisse lülitatud. Vastasel korral ei ole võre trajektoor näha!

Vaata ka

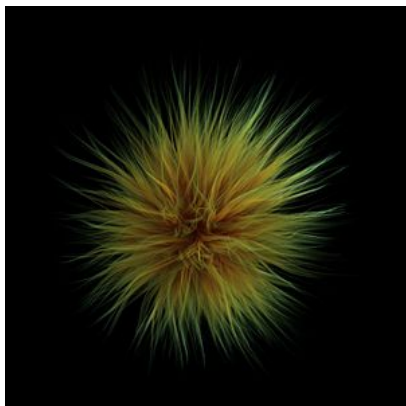
- [Osakesed](#)

Osakesed

Osakesed on suur hulk (harilikult tuhandeid) elemente, mida emiteeritakse võreobjektist. Iga osake võib olla valgustäpp või võre ja osakesed võivad olla omavahel ühendatud või dünaamilised. Nad võivad reageerida paljudele erinevatele mõjudele ja jõududele ja neil on oma eluiga. Dünaamilised osakesed võivad kujutada tuld, suitsu, udu ja muid selliseid asju, nagu näiteks tolm või maagilised loitsud.

[Karva](#) (Hair) tüüpi osakesed on tavaliste osakeste alamklass. Karvasüsteemid moodustavad kiud, mis kujutavad juukseid, karusnahka, muru ja harjaseid.

Sa näed osakesi osakestetõttlejana (Particle Modifier), kuid neid seadistatakse objektikonteksti osakeste alamkontekstis.



Pilt 1: Osakestest tehtud karusnahk ([blend-fail](#)).

Harilikult voolavad osakesed oma võrest ruumi. Nende liikumist võivad mõjutada mitmed asjad, nende hulgas:

- Algkiirus võrest väljudes.
- Emiteerija (tipp, külg või objekt ise) liikumine.
- Liikumine vastavalt "gravitatsioonile" või "õhutakistusele".
- Teiste jõuväljade, näiteks tuule või keeriste mõju, või juhtimine mööda kõverat.
- Vastastikmõju teiste objektidega, näiteks kokkupõrked.
- Osaliselt intelligentsed parve liikmed (karjad, parved jms), mis suhtlevad teiste oma parve liikmetega, kui üritavad jõuda sihtmärgini või vältida kiskjaid.
- Sujuv liikumine pehme keha füüsikaga (ainult karvaosakeste süsteemides).
- Või isegi käsitsi muutmine [sõrestike](#) abil.

Osakesi võib renderdada:

- [Halodena](#) (leegid, suits, pilved).
- Võredena, mida saab omakorda animeerida (näiteks kalad, mesilased jms). Nendel juhtudel iga osake "kannab" teist objekti.
- [Kiududena](#) (*Strands*) ([karvade](#), [karusnaha](#) ja [rohu](#)) jaoks; osakese kogu trajektoor on näha kiuna. Neid kiudusid saab 3D-aknas manipuleerida (kombineerida, lisada, lõigata, liigutada jne).

Igal objektil võib olla mitu osakeste süsteemi. Iga osakeste süsteem võib sisaldada kuni 100 000 osakest. Kindlatel osakeste tüüpidel (karvad ja võtmeosakesed) võib olla kuni 10 000 alamosakest iga osakese kohta (alamosakesed liiguvad ja käituvad enam-vähem nagu nende vastavad ülemosakesed). Praktilisteks piirideks on su mälu maht ja kannatus.

Mitteühilduvus varasemate versioonidega

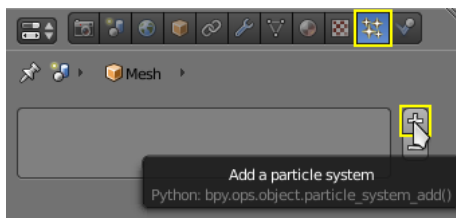
"Vana" osakeste süsteemi, mida kasutati kuni versioonini 2.45 (kaasa arvatud), ja "uue" osakeste süsteemi vahel on mitmeid erinevusi. Nüüd on võimalikud paljud asjad, mida ei saanud teha vana süsteemiga. Uus süsteem on vana süsteemiga kokkusobimatu. Kuigi Blender üritab konverteerida vanu osakeste süsteeme, töötab see ainult osaliselt. Vana süsteem meenutab kõige rohkem uut emiteerija (Emitter) süsteemi (loe edasi, et teada saada, mis see on). Kui kasutad Blender 2.45 või vanemat, [loe vanemat dokumentatsiooni siit](#).

Tööjärjekord

Standardne osakestega töötamise protsess on järgmine:

1. Loo võre, mis hakkab osakesi emiteerima.
2. Loo üks või mitu osakeste süsteemi, mida võre hakkab emiteerima. Sageli mitu osakeste süsteemi mõjuvad vastastikku ja liituvad, et saavutada summaarne soovitud efekt.
3. Sobita iga osakeste süsteemi seaded, et saada soovitud tulemus.
4. Animeeri baasvõre ja teised osakeste võred, mis stseenis osalevad.
5. Määra ja kujunda osakeste trajektoor ja voog.
6. [Karvaosakeste](#) süsteemide puhul: Vooli emiteerija voog (näiteks lõika juuksed paraja pikkuseni ja kammi).
7. Tee lõplik renderdus ja füüsika simulatsioonid ning kohenda, kui vaja.

Osakeste süsteemi loomine

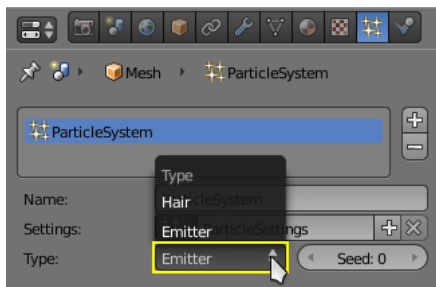


Pilt 2: Osakeste süsteemi lisamine.

Et lisada objektile uut osakeste süsteemi, mine objekti seadete redaktori osakeste (*Particles*) kaardile ja klõpsa väikest + nuppu. Objektile võib olla mitu osakeste süsteemi.

Igal osakeste süsteemil on eraldi seaded. Need seaded võivad olla jagatud erinevate osakeste süsteemide vahel, nii ei pea käsitsi kopeerima iga seadet, vaid sama efekti saab kasutada mitmel objektile. Kui kasutada juhuslikkust (Random), muudetakse omadusi juhuslikult natukene nii, et kui kasutad samu seadeid, paistavad need natuke erinevad.

Osakeste süsteemide tüübid



Pilt 3: Osakeste süsteemide tüübid

Kui oled loonud osakeste süsteemi, täitub omaduste (Property) aken paljude paneelide ja nuppudega. Aga ära satu paanikasse! On kaks erinevat tüüpi osakeste süsteemi ja sa saad nende vahel liikuda rippmenüüga Type (tüüp):

Emitter (emiteerija)

See on vana süsteemiga kõige sarnasem. Sellistes süsteemides emiteeritakse valitud objektist alates alguskaadrist (Start) kuni lõppkaadri (End) osakesi, millel on kindel eluiga.

Karvad

See süsteemi tüüp renderdatakse kiududena ja tal on mõned väga erilised omadused: teda saab 3D-aknas reaalaajas muuta ja lisaks saad kiude animeerida [riidesimulatsiooni](#) (Cloth Simulation) abil.

Seaded osakeste süsteemi paneelil on iga süsteemi tüübi puhul osaliselt erinevad. Näiteks *Pildil 3* on need näidatud ainult emiteerija (Emitter) süsteemi tüübi jaoks.

Ühised valikud

Igal süsteemil on samad põhiseaded, kuid valikud nende sees erinevad sõltuvalt kasutatavast süsteemist. Need seadete kogumid on:

Emission (emissioon)	Seaded osakeste esialgselt jaotumiseks emiteerijal ja viis, kuidas nad stseeni sünnivad.
Cache (puhver)	Et kiirendada reaktsiooniga reaalaajas ja vältida mittevajalikke taasarvutamisi, saab osakese andmeid säilitada mälus või salvestada kettale.
Velocity (kiirus)	Osakeste algkiirus.
Rotation (pöörlemine)	Osakeste pöörlemine.
Physics (füüsika)	Kuidas osakeste liikumine käitub.
Render (renderdamine)	Renderdamise seaded.
Display (esitus)	Reaalaajas kuvamine 3D-vaates.
Children (alamosakesed)	Täiendavate alamosakeste juhtimine.
Field Weights (välja kaalud)	Välise jõudude mõjutugevused.
Force Field Settings (jõuväljade seaded)	Tekitab osakesele jõuvälja.
Vertex Groups (tipugrupid)	Erinevate seadete mõjutamine tipugruppide abil.

Lingid

- [Õppetükid](#)
- [Füüsika puhverdamine ja eeltöötlus](#)
- [Osakeste süsteemi ümberehituse dokumentatsioon](#)
- [Mõtted osakeste süsteemi ümberehituse koodist](#)
- [Staatiliste osakestega karusnahkade kollektsioon](#)

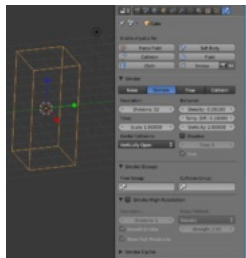
Suitsu simulatsioon

Blenderi uus suitsusimulatsioon põhineb artiklil [turb.php/Wavelet Turbulence for Fluid Simulation](http://turb.php/Wavelet_Turbulence_for_Fluid_Simulation) ja vastaval näidiskoodil, ning on Blenderisse porditud Daniel Genrichi poolt ja töökorras hoiab seda praegu Janne Karhu.

Simulaator põhineb mahulisel vedelikupõhisel mudelil, mille lõpptulemuseks on vokselite võre. Blenderis visualiseeritakse vokselite andmed interaktiivselt, kasutades OpenGL-i varjutamist ja renderdatakse vokseliandmete (*Voxel Data*) tekstuuri abil. Blenderi **suitsusimulatsioon** mähib vokslid ümber olemasolevate [osakeste](#). See vajab osakesi emiteerivat objekti ja 'toimimispiirkonna' objekti, mille sees suits renderdatakse.

See osa dokumentatsioonist vastab versioonile 2.58

Suitsusimulatsioon on samane vedelikusimulatsioonile. Simulatsiooni jaoks on vaja toimimispiirkonna objekti ja voolu objekti.



Suitsu toimimispiirkonna objekt

Nagu vedeliku simulatsioonil, on enamuse seadeid näha, kui toimimispiirkonna objekt on valitud.

Töö käik

- Loo või määra [toimimispiirkonna](#) (*Domain*) objekt või võre, mis määrab simulatsiooni piirid.
- Loo või määra [voolu](#) (*Flow*) objekt, mis määrab, kust suitsu väljutatakse.
- Määra [kokkupõrke](#) (*Collision*) objektid, mis stseenis suitsu mõjutavad.

Toimimispiirkond (*Domain*)

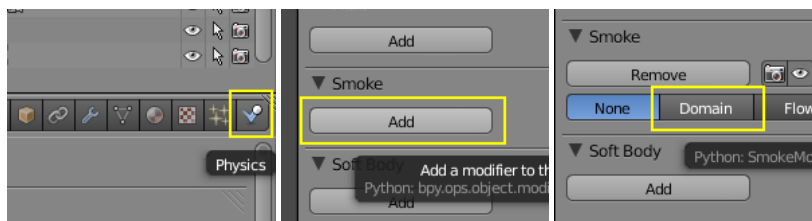
Loo toimimispiirkond

Enne kui saad oma stseenile suitsu lisada, pead määrama ala, kus suitsu simulatsioon toimub. Blenderi füüsikas nimetatakse seda toimimispiirkonnaks (*Domain*). Hea idee on selleks valida kuup, kuna sa saad hiljem selle skaleerida kaamera vaateväljale vastavaks. Meie teeme lihtsal vaikekuubi suuremaks, vajutades kiirklahvi S ja lohistades hiirega.

Ära muuda toimimispiirkonna tipp!

Kui tahad suuremat toimimispiirkonda, siis skaleeri objekti. Selle muutmise muutmisrežiimis põhjustab suitsu ilmutamise renderdamise ajal rohkem kui ühes kohas, nagu korduva tekstuuri puhul.

Veendu, et oled objektirežiimis ja mine füüsika (*Physics*) kaardile. Lisa suits (*Smoke*) ja vali nupp tekstiga 'Domain' (toimimispiirkond). Praeguseks on see kõik, me pöördume väljailmunud uute seadete juurde hiljem tagasi.



Valikud

Resolution (lahutus)

Kui detailne suits on. Lahutusel 32 võtab eeltöötlus mõne sekundi, samas lahutusel 100 võib selleks enamikul PC-del kuluda kuni pool tundi.

Time Scale (ajaskaala)

Mõjutab simulatsiooni esitamise kiirust.

Border Collisions (servakokkupõrked)

Vertically Open (vertikaalselt avatud)

Suits kaob, kui ta puutub vastu toimimispiirkonna põhja või lage.

Open (avatud)

Suits kaob, kui läbib toimimispiirkonna piirid.

Collide All (põrka kõigiga)

Toimimispiirkonna piire koheldakse kui põrkeobjekti, suits põrkub ja jääb toimimispiirkonna sisse.

Temperature and Density (temperatuur ja tihedus)

Kui palju tihedus ja temperatuur suitsu liikumist mõjutavad. Kõrgemad väärtused teevad kiiremini tõusva suitsu.

Vorticity (keerised)

Määrab, kui turbulentne/keerduv või keerlev suits on.

Dissolve (hajumine)

Laseb suitsul aja jooksul hajuda.

Time (aeg)

Suitsu hajumise kiirus.

Slow (aeglane)

Kasutab aja (*Time*) väärtuse asemel väärtust $1/Time$, tehes suitsu hajumise aeglasemaks.

Suitsugrupid**Suits kõrge lahutusega**

Kõrge lahutuse (*High Resolution*) valik lubab sul simuleerida madala resolutsiooniga ja seejärel kasutab müratehnikaid, et lahutust suurendada, ilma seda tegelikult läbi arvutamata. See võimaldab animeerijal valmistada kiiresti ette madala lahutusega simulatsioon ja lisada hiljem detaile, muutmata üldist vedeliku/gaasi liikumist.

Selle jaoks on võimalikud erinevad meetodid, kaasa arvatud vaikemeetod Wavelet, mis realiseerib '[turb.php|Wavelet Vedelikusimulatsiooni turbulentsi](#)' meetodi

Resolution/Divisions (lahutus/jagamisi)

Suurendab suitsu lahutust müra abil selle teguri võrra.

Smooth Emitter (sile emiteerija)

Silub emiteeritud suitsu, et vältida suitsu kandilisust.

Show High Resolution (näita kõrglahutust)

Näitab kõrget lahutust, kasutades võimendust.

Noise Method (müra meetod)

Wavelet

FFT (Fourier)

Strength (tugevus)

Müra tugevus.

Suitsuvälja kaalud

Määrab, kui palju erinevad jõud ja jõuväljad suitsu mõjutavad.

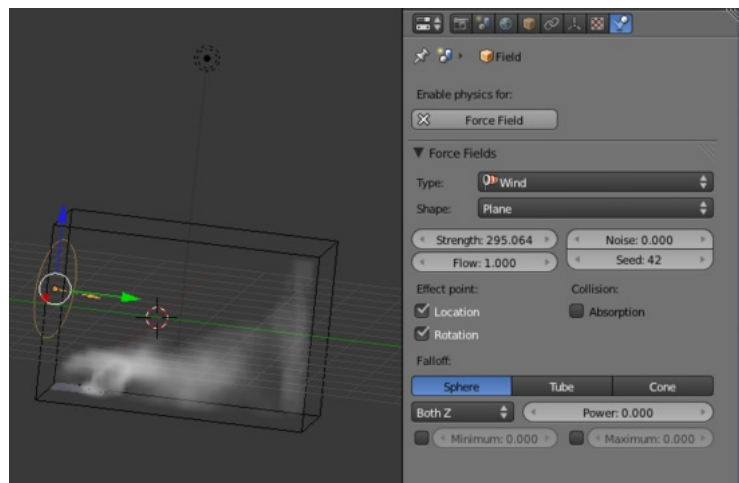
Gravity (gravitatsioon)

Kui palju gravitatsioon suitsu mõjutab.

All (kõik)

Muudab kõikide jõuväljade üldist mõju.

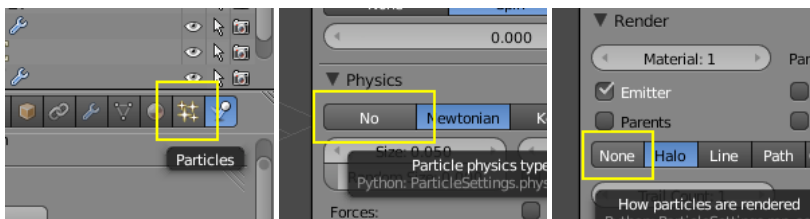
Ülejäänud seaded määravad, kui palju erinevad jõuväljad suitsu mõjutavad.

**Sissevool (Inflow)****Loo voolu objekt**

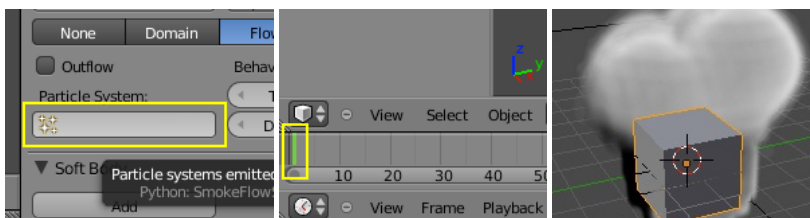
Nüüd, kui me oleme määranud mahu, mille sees suits hakkab olema, lisame objekti, millest suitsu hakatakse emiteerima. Lisa teine kuup ⇨ ShiftA ⇨ Mesh ⇨ Cube (võre->kuup) ja veendu, et see oleks toimimispiirkonna kuubi sees; valitud peab olema 3D-vaade).

Olles muutmisrežiimis, mine füüsika kaardile ja lisa väikesele kuubile samuti suits. Seekord vali vool (Flow).

Suitsu ei emiteerita mitte objektist endast, vaid osakestest, mida objekt emiteerib. Seega, peame looma osakeste süsteemi. Väike kuup ikka veel valitud, mine osakeste kaardile. Lisa uus osakeste süsteem ja lülita välja füüsika, sest me soovime, et suits emiteeritaks liikumatust kohast. Samuti ei taha me näha osakesi, seetõttu lülita välja ka renderdamine.



Nüüd mine tagasi füüsika kaardile ja vali osakeste süsteem suitsu alajaotuses. Seal peaks olema süsteemide nimekiri, kust saab hetkel valida ainult ühe süsteemi, mille nimi on 'ParticleSystem', sest me ei muutnud nime. Nüüd saad kerida mööda ajatelge, et näha kuubikust suitsu tõusmas. Teine võimalus suitsu eelvaateks on animatsiooni mängimine kiirklahviga AltA (peata mängimine sama klahviga).



Seaded

Outflow (väljavool)

Kustutab simulatsioonist suitsu.

Particle System (osakeste süsteem)

Sellest objektist emiteeritud osakeste süsteem.

Initial Velocity (algkiirus)

Suits saab oma kiiruse emiteerija osakestelt.

Multiplier (kordaja)

Kordaja suitsule edasiantava kiiruse seadmiseks.

Algväärtused

Absolute Density (absoluutne tihedus)

Lubab emiteerija alas ainult etteantud tihedust.

Density (tihedus)

Algne tiheduse väärtus.

Temp. Diff. (temperatuuride erinevus)

Temperatuuri erinevus võrreldes ümbritseva temperatuuriga.

Kokkupõrked

Suits võib põrkuda teiste objektidega, kui kasutada tema 'Põrke' (*Collision*) valikut. Hetkel on toetatud ainult põrked staatiliste objektidega.

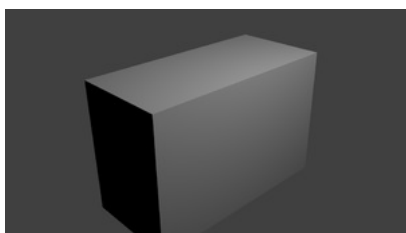
Jõud

Töötavad ka Blenderi jõuväljad (nagu tuul ja keerisväljad), mis muudavad suitsu simulatsiooni samamoodi nagu teisi füüsikalisi süsteeme, näiteks osakesi.

Suitsu renderdamine

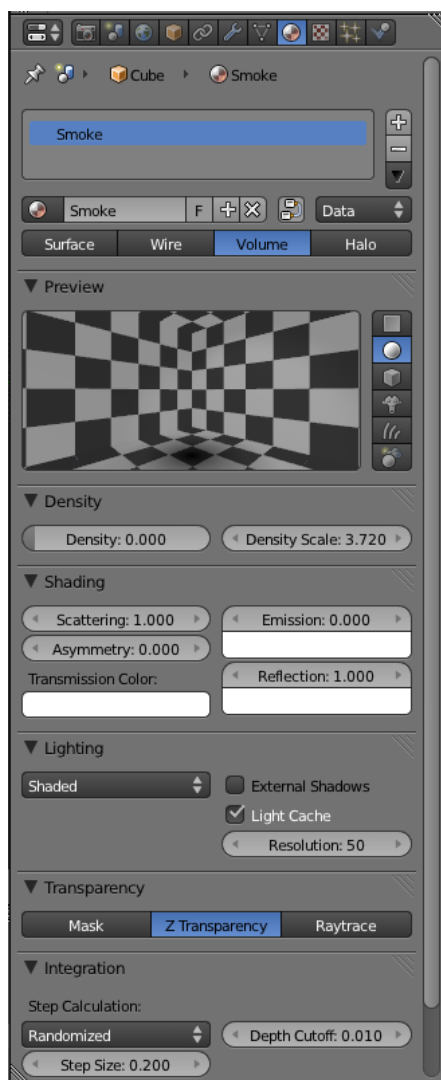
Loo materjal

Suitsu simuleerimine on lihtne, renderdamine samas aga mitte.

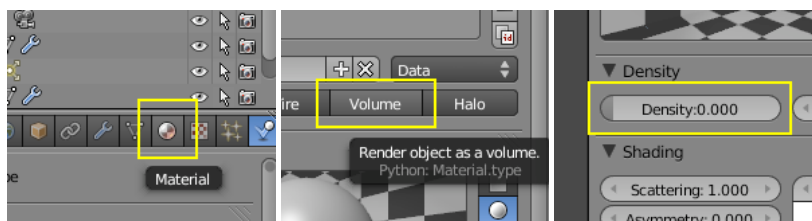


Praegu renderdamine annab tulemuseks lihtsalt suure risttahuka (nagu pildil, F12) või kokkujooksmise (animatsioon, CtrlF12).

. Materjal peab olema mahuline materjal tihedusega (*Density*) 0 ja suure tiheduse skaala (*Density Scale*) väärtusega.

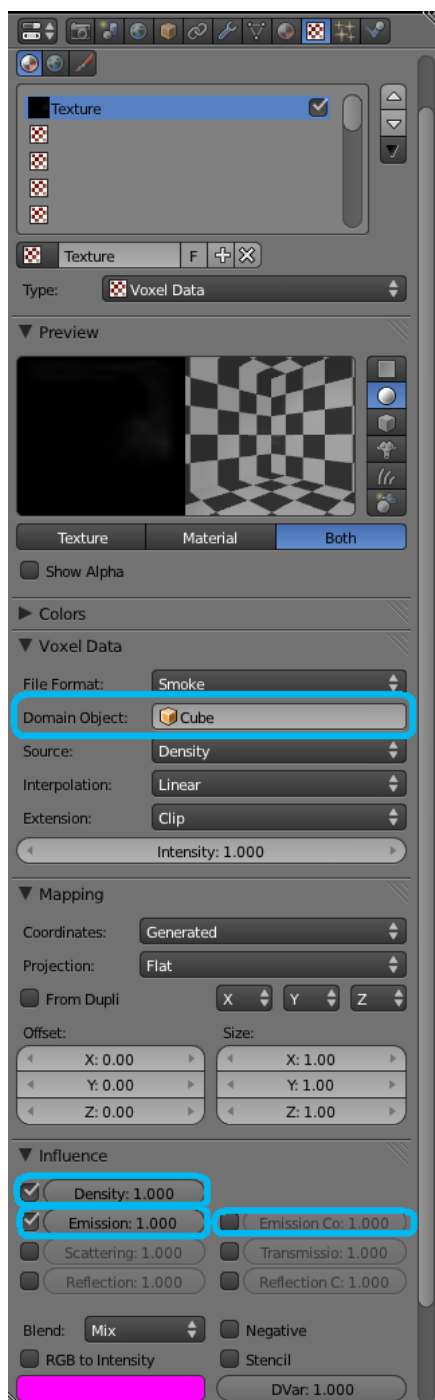


Esimese probleemi saab kergesti lahendada, kui määrata toimimispiirkonna kuubi jaoks õige materjal ja tekstuur. Suits vajab keerukat materjali, et õigesti renderduda. Vali suur kuup ja mine materjalide kaardile. Seal määra materjali tüübiks *Volume* (mahuline) ja sea tiheduseks (*Density*) 0. Kui sead tiheduse väärtuse suuremaks kui 0, täitub toimimispiirkonna kuup üleni mahulise materjaliga. Samas mõjutavad suitsu ka [teised seaded](#). Me käsitleme neid hiljem.



Lisa tekstuur

Lisaks vajab suits ka oma teksturi. Blender 2.5-l on uus tekstuur just suitsu renderdamiseks, nimelt [voksliaandmed \(Voxel Data\)](#). Sa ei tohi unustada määrata toimimispiirkonna objekti ja muuta mõju.

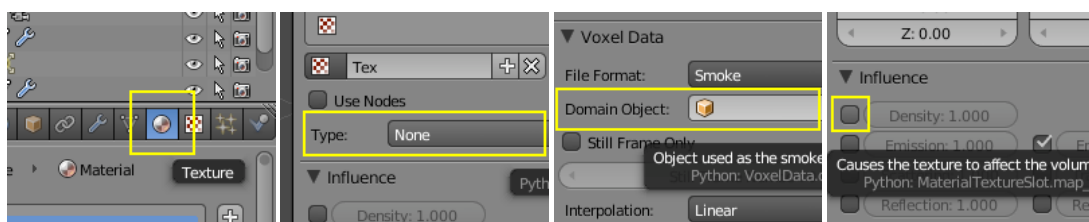


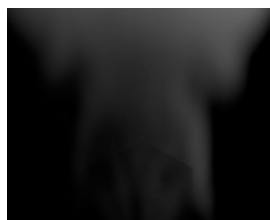
Mine teksturi kaardile ja määrä tüübiks *Voxel Data* (voksliandmed). Voxel Data seadete alt vali toimimispiirkonna (*Domain*) objektiks meie toimimispiirkonna kuup (selle nimi peaks olema lihtsalt *Cube*, kuna kasutame Blenderi vaikekuupi. *Mõju (Influence)* alt lülita sisse 'Tihedus' (*Density*) ja jäta selle väärtuseks 1000 (emissioon (*Emission*) peaks olema samuti automaatselt sisse lülitatud). Nüüd peaksid saama renderdada üksikuid kaadreid. Samuti saad valida suitsu värvi, lülitades "emissiooni värvi" (*Emission Color*) tagasi sisse.



Et näha suitsu selgemalt

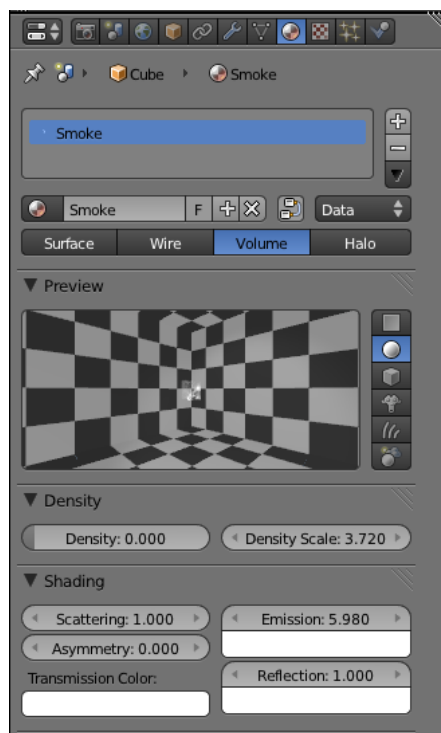
Vali maailma (*World*) kaardilt horisondi jaoks väga tume värv.



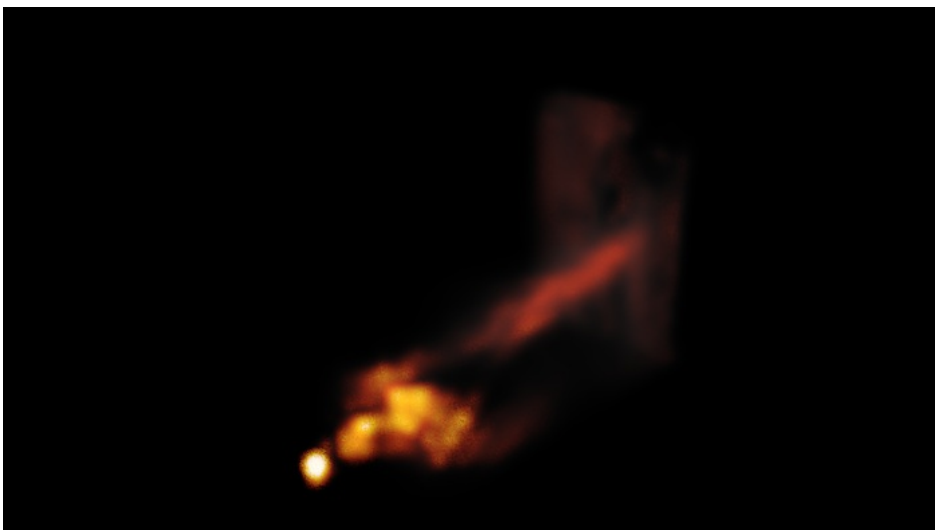
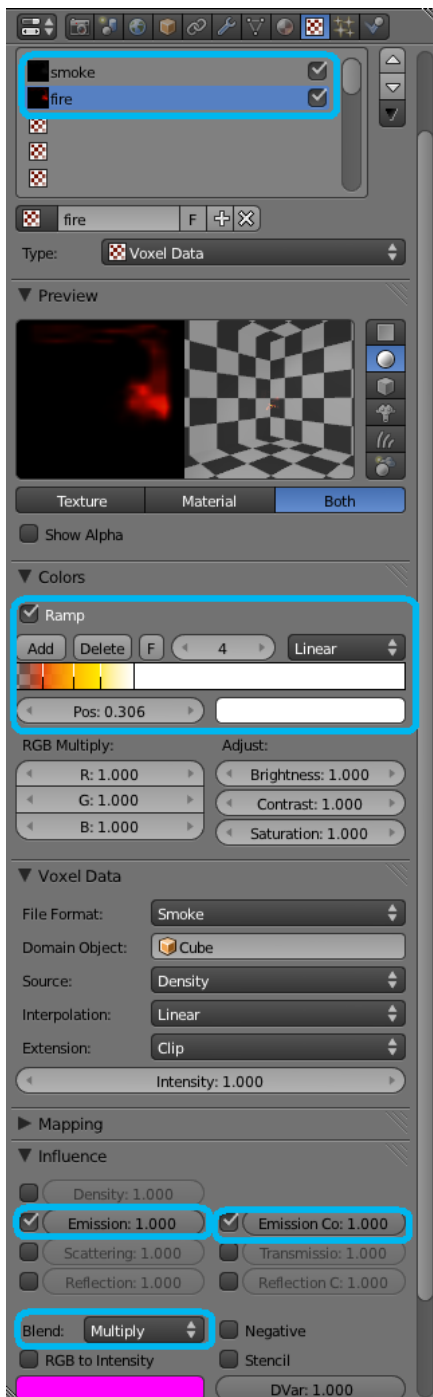


Suitsusimulaatori laiendamine: Tuli!

Saad muuta oma suitsu teise tekstuuri abil tuleks! Et teha tuld, keera materjalide paneelil emissiooni (*Emission*) väärtust juurde.



Siis lisa teine tekstuur. (Jäta vana tekstuur alles, muidu ei ole suitsu näha.) Anna sellele leekiv värviüleminek - katvusega (*alpha*) värvide põhjal ja muuda mõju emissiooniks (*Emission*) ja emissiooni värviks (*Emission Color*). Vali segamise (*Blend*) tüübiks Multiply (*korrutamine*).

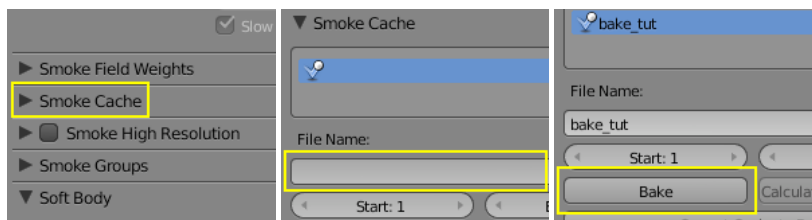


Suitsu simulatsioonide eeltöötlus

Kui soovid teha animatsiooni, pead oma suitsu esmalt eeltöötlema. Eeltöötlus on lihtsalt simulatsiooni arvutamine. Et suitsu eeltöödelda, peab fail olema salvestatud. Arvutused salvestatakse vahemälu faili, millele saab ka nime anda. Mõnedel juhtudel võib Blender eeltöötlust tehes kokku joosta. [Vaata tõrkeotsingut]

Kerides mööda ajatelge või jooksutades animatsiooni 3D-vaates kiirklahviga **AltA**, oled sa juba teinud mälu reaalajas eeltöötlust. Aga et animatsiooni renderdada, peavad eeltöödeldud andmed olema kettal. Ja enne kui saad eeltöödelda, pead oma .blend-faili salvestama.

Järgmiseks vali toimetamispiirkonna kuup, mine füüsika (*Physics*) kaardile ja ava suitsu vahemälu sektsioon (*Smoke Cache*). Anna oma vahemälu failile nimi, sisestades see tekstikasti ja vajutades ↵ Enter. Vajutades nupule *Bake* (eeltöötle), arvutatakse simulatsiooni andmed ja salvestatakse kettale. Kas märkad, et ajatelje viga on kadunud? Nüüd peaksid sa saama animatsiooni renderdada.



Tõrkeotsing

K: Blender jookseb kokku, kui vajutan suitsu simulatsiooni nuppu *Domain* (toimetamispiirkond)!

V: Selle põhjustavad vananenud draiverid. Uuenda oma draivereid.

K: Blender jookseb suitsu simulatsiooni eeltöödeldes kokku!

V: Sul saab selle arvutamisel operatiivmälu otsa. Proovi eeltöödelda madalama lahutusega.

K: Suits ei renderdu!

V: Mine tagasi ja loe dokumentatsiooni.

K: Kui proovin teha tuld, annab see imelikke tulemusi või ei paista üldse välja.

V: Veendu, et sul on materjalile määratud kõrge emissiooni (*Emission*) väärtus, et sul on suitsu tihedustekstuuri ja et oled tule tekstuuri segamisviisiks määranud *Multiply* (korruta).

Välised lingid

- [Põhjalik sissejuhatus Blender 2.5 suitsu ja tule meetoditesse koos tavaliste vigade selgitusega](#)
- [MiikaH õpetus, kuidas teha Blenderis realistlikku tuld](#)

Pehmed kehad



Pilt 1a: Pehme keha kujutamas riiet, mille alt ilmub välja tekst. [Animatsioon](#) – [.blend-fail](#)

Pehme keha (*Soft Body*) mõistega tähistatakse kas pehme või jäiga moonutatava keha simulatsiooni. Blenderis on see süsteem parim lihtsate riideobjektide ja suletud võrede jaoks. [Riide simulatsiooni](#) füüsika kasutab teistsugust lahendajat ja on riide jaoks parem.

Simulatsioon teostatakse, rakendades objekti tippudele või kontrollpunktidele jõudusid. On välised jõud, nagu gravitatsioon või jõuväljad, ja sisemised jõud, mis hoiavad tippe omavahel koos. Nii saad simuleerida vorme, mida objekt võtaks reaalsuses, kui tal oleks maht ja ta oleks millegagi täidetud ning talle mõjuksid reaalsed jõud.

Pehmed kehad võivad teiste objektidega vastastikku üksteist mõjutada (kokkupõrked, *Collisions*). Nad saavad mõjutada ka iseennast (kokkupõrge iseendaga, *Self Collision*).

Pehme keha simulatsiooni tulemust saab konverteerida staatiliseks objektiks. Samuti saad sa *muuta eeltöötlust*, st muuta vahepealset simulatsioonitulemust ja sealt alates simulatsiooni uuesti jooksutada.

Pehme kehade kasutamise tüüpiline stsenaarium



Pilt 1b: Tuulekott. Kott on pehme keha, nagu ka nõõrid. [Animatsioon](#) – [.blend-fail](#)

Pehmed kehad sobivad hästi järgnevate asjade jaoks:

- Elastsed objektid nii kokkupõrgetega kui ka ilma.
- Lippude ja kangaste reageerimine jõududele.
- Teatud modelleerimisülesanded, nagu padi või laudlina laual.
- Blenderil on veel teine simulatsioonisüsteem riietuse jaoks (vaata [riiete peatükki](#)). Aga sa saad mõnikord kasutada pehmeid kehasid teatud riide osade jaoks nagu laiad varrukad.
- Karvad (juhul kui sa minimeerid põrked).
- Kiikuvate kõite, kettide ja muu taolise animatsioon.

Järgnev video võib anda sulle veel mõned ideed: [\[1\]](#), [\[2\]](#)

Pehme keha loomine

Pehme keha simulatsioon töötab kõigi objektidega, millel on tipud või kontrollpunktid:

- Võred
- Kõverad
- Pinnad
- Sõrestikud

Et aktiveerida objektile pehme keha simulatsioon:

- Mine objektikonteksti (F7) ja vali füüsika (Physics) alamkontekst.
- Aktiveeri nupp Soft Body (pehme keha).

Ilmub palju valikuid. Et saada aimu kõikidest seadetest, vaata [seda lehekülge](#).

- Pehme keha simulatsiooni alustamiseks vajuta AltA.
- Pausiks vajuta Space, jätkamiseks AltA.
- Simulatsiooni peatab Esc.

Simulatsiooni kvaliteet

Seaded paneelil Soft Body Solver (pehme keha lahendaja) määravad simulatsiooni täpsuse.

Min Step (minimaalne sammude arv)

Minimaalne simulatsioonisammude arv kaadri kohta. Suurenda seda väärtust, kui pehme keha ei taba kiiresti liikuvaid pörkeobjekte.

Max Step (maksimaalne sammude arv)

Maksimaalne simulatsioonisammude arv kaadri kohta. Harilikult seatakse simulatsiooni sammude arv dünaamiliselt (Error Limit - vea piir), aga sul on arvatavasti hea põhjus selle muutmiseks.

Auto-Step (automaatsed sammud)

Määrab sammude suurused automaatselt kiiruste abil.

Error Limit (vea piir)

Määrab tulemuse üldise kvaliteedi. Vaikeväärtus 0.1. Kõige kriitilisema tähtsusega seade, mis ütleb, kui täpselt lahendaja peaks pörkeid kontrollima. Alusta väärtusest, mis on 1/2 keskmisest serva pikkusest. Kui on nähtavaid vigu, võidainad või ülemääraseid reaktsioone, vähenda selle väärtust. Lahendaja jälgib, kui "halvasti" tal läheb ja seade Error Limit (vea piir) laseb lahendajal teha mõned "adaptiivsed sammu suuruse muudatused".

Fuzzy (hägune)

Simulatsioon on kiirem, aga vähem täpne.

Choke (pidurda)

Pidurdab (vähendab väljumiskiirust) tippe või servi, mis pörke võret läbistavad.

Diagnostika

Print Performance to Console (trüki jõudlus konsoolile)

Trükitab konsoolile, kuidas lahendaja töötab.

Estimate Matrix (hinda maatriksit)

Hindab maatriksit. Jagab maatriksi asukohaks, pöördeks ja skaalaks.

Vahemälu ja eeltöötlus

Pehmed kehad ja teised füüsikalised simulatsioonid kasutavad eeltöötlusteks ja vahemäluks ühtset süsteemi. Vaata selle ülevaadet peatükist [osakeste vahemälu](#).

Simulatsiooni tulemused salvestatakse animatsiooni esitamisel automaatselt kettale, nii et järgmine kord seda käivitades loeb ta tulemusi kettalt ja mängib kiiremini. Kui sa eelarvutad (Bake) simulatsiooni, siis on vahemälu kaitstud ja sult küsitakse kinnitust, kui proovid muuta seadeid, mis teevad vajalikuks ümberarvutamise.



Ole ettevaatlik seadetega Start (algus) ja End (lõpp)

Simulatsioon arvutatakse ainult alguse (Start) ja lõpu (End) kaadrite vahel (Bake (eeltöötluste) paneelil), isegi kui sa tegelikult simulatsiooni ei eeltöötle! Seega, kui tahad pikemat simulatsiooni kui vaikimisi määratud 250 kaadrit, pead muutma kaadri End (lõpu) väärtust.

- Vahemälu kasutamine
- Kui animatsiooni esitatakse, kirjutab iga füüsikaline süsteem kõik alguskaadri ja lõpukaadri vahelised kaadrid kettale. Need failid salvestatakse .blend-failiga koos alamkaustadesse, mille nimed algavad eesliitega "blendcache".
 - Vahemälu tühjendatakse automaatselt, kui teed muudatusi - kuid mitte kõigi muudatuste puhul ja seega on mõnikord vaja seda käsitsi tühjendada. Näiteks kui oled teinud muudatusi jõuväljades. NB! Vahemälu täitmiseks peab taasesitus algama enne, või samast kaadrist, kust simulatsioon algab.
 - Kui sul ei lubata vajalikku alamkataloogi kirjutada, vahemälu salvestamist ei toimu.
 - Vahemälu saab vabastada iga füüsikalise süsteemi kohta eraldi paneelil oleva nupuga või kiirklahviga CtrlB kiirklahviga, mis vabastab kõigi valitud objektide vahemälu.
 - Sul võib tekkida probleeme, kui su .blend-faili asukoha aadress on liiga pikk ja su operatsioonisüsteemil on faili täisnime pikkus piiratud.
- Eeltöötlus
 - Pärast eeltöötlust on süsteem muudatuste eest kaitstud.
- Eeltöötluste (Bake) tulemused tühistatakse kõigi valitud objektide puhul kiirklahviga CtrlB või klõpsates nupule Free Bake (tühistab eeltöötlust) üksiku pehme keha jaoks.
 - Kui võret muudetakse, ei arvutata simulatsiooni uuesti.
- - Renderfarmide puhul on kõige parem kõik füüsikalised süsteemid eeltöödelda ja kopeerida blendcache-kaust samuti renderfarmi.

Suhtlus reaalaajas

Pehme keha simulatsiooniga töötamiseks on käepärane kasutada ajatelje (Timeline) akent. Saad vahetada kaadreid ja simulatsioon on alati näha tema tegelikus olekus. Valik Continue Physics (jätk füüsikat) ajatelje akna menüüs Playback (taasesitus) lubab sul simulatsiooniga suhelda reaalaajas, st liigutada pörkeobjekte või raputada pehme keha objekti. Ja see on päris lõbus!



Continue Physics (jätk füüsikat) ei tööta, kui mängid animatsiooni kiirklahviga AltA

See töötab ainult juhul, kui mängid animatsiooni ajatelje (Timeline) aknas nupuga Play.

Siis saad simulatsiooni jooksutamise ajal pehme keha objekti valida ja rakendada (Apply) töötleja muutmiskonteksti töötlejate

(Modifiers) paneelililil. See teeb deformatsiooni püsivaks.

Vihjed

- Pehmed kehad toimivad eriti hästi, kui tippude jaotus on ühtlane. Hea pörke jaoks on vaja piisavalt tippe. Sa muudad moonutust (jäikust), kui lisad kindlatesse piirkondadesse rohkem tippe (vaata animatsiooni pildil *Pilt 1b*).
- Pörke arvutamine võib võtta kaua aega. Kui mingi asi ei ole näha, siis milleks seda arvutada?
- Et kiirendada pörke arvutamist, on sageli kasulik lasta objektidel pörgata täiendava lihtsama, nähtamatu, mõnevõrra suurema objektiga (vaata näidet pildil *Pilt 1a*).
- Kasuta pehmeid kehasid ainult siis, kui sellel on mõtet. Kui üritad katta keha võre riidetükiga ja animeerida ainuüksi pehme kehana, ei tule sellest midagi välja. Pehme keha karvade pörge iseenestega on võimalik aktiveerida, kuid seal pead sa juba üksi tee leidma. Me tegeleme [pörgetega](#) hiljem üksikasjalikumalt.
- Proovi ja kasuta sõrestiku (Lattice) või suunava kõvera (Curve Guide) pehmeid kehasid objekti enda asemel. See võib olla kordades kiirem.

Lingid

- [Arendaja märkmed](#)
- [Ahela kiikumine](#)
- [Pehmed kehad skeletiga tegelaste jaoks](#)

Sissejuhatus

Valgustus on renderdamise juures väga oluline teema, olles võrdne modelleerimise, materjalide ja tekstuuridega. Ka väga detailselt modelleeritud ja tekstuuritud stseen jätab ilma korraliku valgustuseta kehva mulje, samas muudab oskuslik valgustus ka lihtsa mudeli väga realistlikuks. Kahjuks jätab kogematu kasutaja valgustuse tihti peale kahe silma vahele ning arvab, et kui reaalses maailmas on ruum tavaliselt valgustatud ühe valgustiga (lamp, päike vms), piisab ka arvutigraafikas ühest valgusallikast. See aga pole nii, sest reaalses maailmas peegeldub sellest ühest allikast kiirgunud valgus tema teele jäävate esemete pindadelt tagasi ning muudab varjud pehmeks – pimedad alad ei ole täiesti süsimustad, vaid kergelt valgustatud.

Tajumise piirangud

Esemete värvi ning stseeni valgustust mõjutavad järgmised tegurid:

- Sinu võime värve eristada (osaline värvipimedus on väga levinud).
- Pildi vaatamiseks kasutatav seade või kandja (näiteks LCD-monitor vs läikpaberil trükipilt).
- Pildi kvaliteet (näiteks JPEG pakkimisaste **0.4** või **1.0**).
- Keskkond, kus sa pilti vaatad (näiteks läikiva ekraaniga kineskoopmonitor hämaras toas või päikeselises, siniseks värvitud ruumis).
- Sinu aju [omadus tajuda](#) heledust ja värvi erinevalt sõltuvalt taustobjektidest ja tagaplaanist

See tähendab, et sama pilt võib isikule A toas B ja monitorilt C paista oluliselt erinev kui ta paistab isikule D, kes vaatab seda metroos E ja printitud paberilt F.

Globaalsed mõjud

Blenderis saab muuta järgmisi valgustust mõjutavaid sätteid:

- Maailma (*World*) [üldvalguse](#) (*Ambient Light*) värvus.
- [Kaudvari](#) (*Ambient Occlusion*) on üks viisidest, kuidas simuleerida üldvalguse langemist objektile.
- Millisel määral üldvalgus muudab objekti [materjali](#) värvi.
- [Kaudne valgustus](#) (*Indirect Lighting*), mille puhul ühe objekti värv kiirgub teisele objektile.
- Kasutatav renderdusmootor (Blenderi sisene renderdusmootor või [Yafray](#)).
- Stseenis kasutatavad [valgusallikad](#).

Reaalses maailmas toimuvat valguse põrkumist simuleerivad kaudvari (maailma seade), puhvervarjud (mis ligilähedaselt hindavad objektide poolt heidetavaid varje) ja kiirtejälitus (mis järgib valgusallikast lähtuvate footonite teekonda). Blenderis on võimalik kasutada ka [kaudset valgustust](#) (Indirect Lighting). Kiirtejälituse, kaudvarju ja kaudse valgustuse kasutamine nõuab arvatult palju ressursse. Blender töötab kõige kiiremini oma sisseehitatud laotusjoonerenderdajaga, mis on tegelikult väga hea renderdaja. Selline renderdusmootor on väga kiire, kuna ta ei proovi simuleerida valguse tegelikku käitumist, vaid kasutab paljusid lihtsustavaid hüpoteese.

Valgustuse sätted

Lisaks mainitud globaalsetele mõjudele lisandub stseeni valgus seal olevatest lampidest. Peamised valgustite sätted, mida sa saad juhtida, on järgmised:

- Kasutatav valgusti tüüp: (päikesevalgus (Sun), kohtvalgus (Spot), lamp (Lamp), poolkera (Hemi) jne).
- Valguse värvus.
- Valgusti paigutus ja suund.
- Valgusti enda sätted, sealhulgas valgustugevus ja selle nõrgenemise viis.

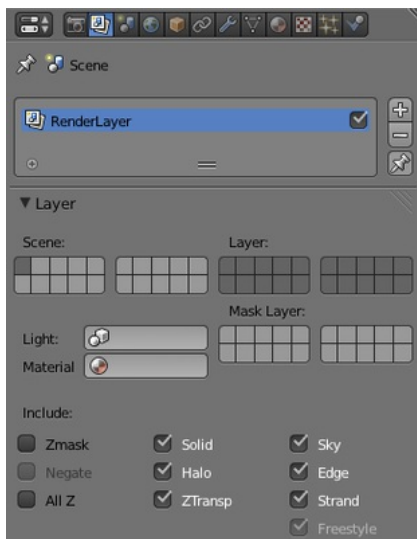
Seejärel mõjutab valgustust see, kuidas objekti materjali [varjutaja](#) reageerib valgusele.

Järgnev peatükk selgitab kõike eelpool mainitud ning näitab, kuidas lambid saavad stseeni valgustamisel toimida ühtse süsteemina. Selles peatükis kirjeldame põhjalikumalt Blenderi erinevaid valgustitüüpe ja nende omadusi, nii tugevaid kui ka nõrku külgi. Näidetena kirjeldame ka erinevaid valgusparke, kaasa arvatud klassikalist kolme punkti valgustust.

Töö valgusega

Selles käsiraamatus oleme valgustuse peatüki paigutanud materjalide osa ette, seega tuleks valgustus üles seada enne võredele materjalide määramist. Kuna materjalide varjutusalgoritmid käituvad vastavalt valguse omadustele, ei näe materjalid ilma kohase valgustuseta õiged välja. Sellisel juhul võid sa püüda lõpmatuseni tulutult materjali varjutajat seadistada, sest tegelik probleem on hoopis kehvast valgustuses. Mitte ühelgi selle peatüki näidispildil ei ole kera, kuubi ega tausta materjal määratletud.

Materjalide tühistamine valgustuse algseadistamiseks



Materjali lahter Material renderduskihtide paneelil Render Layers

Kui sa oled juba objektile materjalid määranud ja asusid valgustust üles seadma pärast seda, soovitame luua üldise halli materjali ilma tipu värvi (Vertex Color), küljetekstuuri (Face Color) ja varjutajata (Shader) – lihtsalt tavaline neutraalhall värv RGB väärtustega **(0.8, 0.8, 0.8)**.

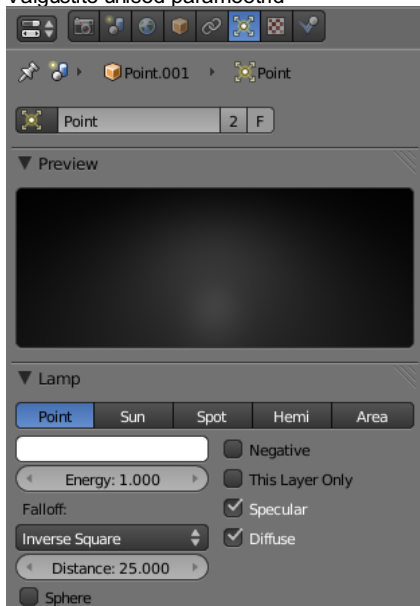
Kui see on tehtud, mine renderduskonteksti Render. Renderduskihtide Render layers paneelis vali välja materjal Material väärtuseks "Grey". See tühistab kõik määratud materjalid ja renderdab kõik objektid seda igavat halli värvi. Nüüd võid seda materjali kasutades valgustuse üles seada. Algsete materjalide taastamiseks lihtsalt tühjenda see väli.

Valgusallikad

Nagu juba mainitud, on Blenderis võimalik kasutada mitut erinevat valgustusviisi, nagu näiteks kaud- või üldvalgus. Siiski on kõige tihedamini kasutusel "valgustid" või "lambid". Selles osas räägime valgustite omadustest üldisemalt (täpsemat infot iga lambitüübi kohta leiab osast [Lambid](#)):

- [Valguse omadused](#) – kõigile lambitüüpidele ühised sätted.
- [Valguse nõrgenemine](#).
- [Tekstuurid](#) – kuidas anda valgustitele tekstuure.
- [Mida valgus mõjutab](#).
- [Valgustid teistes kontekstides](#) – valgustitega seotud parameetrid teistes kontekstides.

Valgustite ühised parameetrid



Valgusti parameetrite paneel Lamp Properties

Blenderis on viis erinevat valgusti tüüpi. Järgnevad seadistused on neil kõigil ühised:

Objekti andmed (*Object Data*)

Valgusti objektiandmete sirvija (Browse Light Object Data)

Klõpsa, et valida üks valgusti kõigi stseenis olevate hulga.

Nimi (Name)

Parajasti valitud valgustiobjekti nimetus. Valgusti nime vahetamiseks muuda seda välja.

Kasutajate arv (Number of Users)

Valgustite arv, mis kasutavad samu objektiandmeid.

F

Loo nende objektiandmete jaoks võltskasutaja.

Eelvaade (*Preview*)

Valgusti seadete kiire eelvaade.

Valgusti (*Lamp*)

Kaugus (Distance)

Kauguse väli Distance määrab, kui mitme Blenderi ühiku (BU) kaugusel on antud valgusallika valgustugevus kahanenud pooleni algsest. Objektid, mis on valgustile lähemal, saavad rohkem valgust, kaugemad objektid vähem. Erinevad valguse nõrgenemiskõverad ja osad seaded mõjutavad seda, kuidas kauguse välja tõlgendatakse. Sellest tulenevalt ei käitu ta alati täpselt ühtmoodi, vaata ka peatükki [valguse nõrgenemisest](#).

Valgustugevus (Energy)

valgusallikast kiirgava valguse tugevus (väärtusega "0.0" kuni "10.0").

Värvus (Color)

valgusallikast kiirgava valguse värvus. Avab värvivalija.

Negatiivne (Negative)

Muudab valgusti kiirguse negatiivseks.

Ainult sama kihi objektid (This Layer Only)

Valgustatakse ainult objekte, mis on valgustiga samal kihil.

Läige (Specular)

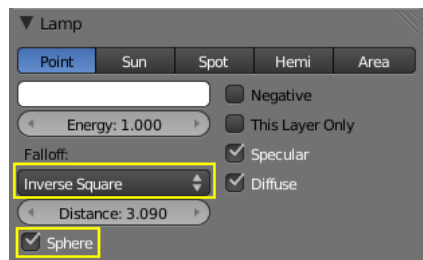
Valgusti tekitab objektidel läiget.

Hajusvalgus (Diffuse)

Valgusti mõjutab hajusvarjutust (*Diffuse Shading*).

Valguse nõrgenemine

Kirjeldus



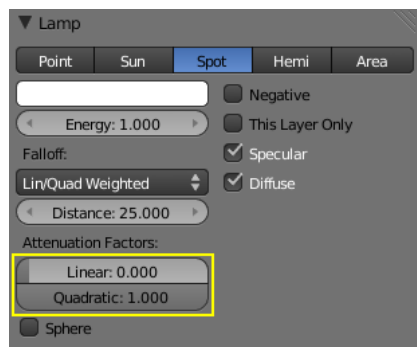
Valgusti paneel Light, valguse nõrgenemiskõvera valikud esile tõstetud

Punktvalgusel (Lamp) ja kohtvalgusel (Spot) on kaks peamist parameetrit valguse nõrgenemise kontrollimiseks.

- Valgusti valguse nõrgenemiskõvera rippmenüü Falloff
- Kera nupp Sphere

Valguse nõrgenemiskõverad

Lineaarne/ruutsõltuvus (*Lin/Quad Weighted*)



Valgusti paneelis Light on esile toodud lineaarne/ruutsõltuvuses nõrgenemiskõver Lin/Quad Weighted Falloff.

Selle variandi korral on paneelil kaks liugurit: lineaarne (Linear) ja ruutsõltuvus (Quadratic), mille abil saab määrata vastavalt valguse nõrgenemiskõvera "lineaarsust" ja "ruutsõltuvust".

See nõrgenemiskõvera tüüp lubab kahe nõrgenemisgraafiku (lineaarse ja ruutsõltuvuse) omavahelist kombineerimist.

Lineaarne (*Linear*)

Liuguri väärtus võib olla **0.0** kuni **1.0**. Lineaarse liuguri Linear väärtusega **1.0** ja ruutsõltuvuse liuguri Quadratic väärtusega **0.0** annavad kokku täiesti lineaarse nõrgenemiskõvera valguse. See tähendab, et jõudes kauguse väljal Distance määratud arvu Blenderi ühikute kaugusele valgusallikast on valgusjõud vähenenud täpselt poole võrra.

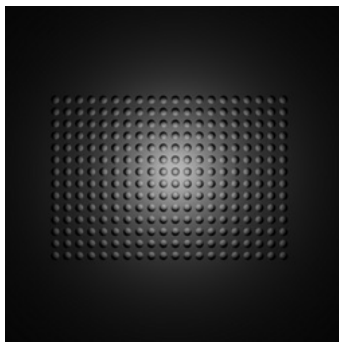
Kui ruutsõltuvuse liuguri Quadratic väärtus on **0.0**, siis arvutatakse valguse nõrgenemine järgmise lineaarse valemi järgi:

$$I = E \times (D / (D + L \times r))$$

Kus

- I on arvutatud valgustugevus vastavas punktis.
- E on valgusjõu liuguri Energy väärtus.
- D on kauguse välja Distance väärtus.
- L on lineaarse liuguri Linear väärtus.
- r on selle punkti kaugus valgusallikast.

Ruutsõltuvus (*Quadratic*)



Lineaarse/ruutsõltuvuses
nõrgenemiskõvera Lin/Quad
Weighted Falloff vaikesätetega
valgusti.

Ka selle liuguri väärtus võib olla **0.0** kuni **1.0**. Ruutsõltuvuse liugur Quadratic väärtusega **1.0** ja lineaarse liugur Linear väärtusega **0.0** annavad kokku täiesti ruutsõltuvusliku nõrgenemiskõvera valguse.

Ruutsõltuvuslik valguse nõrgenemine vastab reaalses maailmas toimuvale valguse nõrgenemisele täpsemalt kui lineaarne. Ruutsõltuvuslik valguse nõrgenemine on ka lineaarse/ruutsõltuvuses nõrgenemiskõvera Lin/Quad Weighted Falloff vaikeseadeks (vaata ka pilti *Lineaarse/ruutsõltuvuses nõrgenemiskõvera Lin/Quad Weighted Falloff vaikesätetega valgusti*).

Nagu lineaarsel nõrgenemiskõveral, on ka siin valgusallika valgustugevus vähenenud täpselt pooleni, kui kaugus valgustist on lahtris Distance määratud arv Blenderi ühikuid. Võrreldes ruutsõltuvuslikku nõrgenemiskõverat lineaarsega, näeme, et ruutsõltuvusliku nõrgenemiskõveraga valgusti valgustugevus langeb aeglasemalt enne kauguse väljal määratud väärtuseni jõudmist, sealt edasi aga palju kiiremini.

Kui lineaarse liuguri Linear väärtus on **0.0**, siis arvutatakse valguse nõrgenemine järgmise ruutsõltuvusliku valemi järgi:

$$I = E \times (D^2 / (D^2 + Q \times r^2))$$

Kus

- I on arvutatud valgusallika valgustugevus vastavas punktis.
- E on valgusjõu liuguri Energy väärtus.
- D on kauguse välja Distance väärtus.
- Q on ruutsõltuvuse liuguri Quad väärtus.
- r on selle punkti kaugus valgusallikast.

Lineaarse ja ruutsõltuvusliku segamine

Kui mõlemad liuguri (Linear ja Quad) väärtus on suurem kui **0.0**, kasutatakse valguse nõrgenemiskõvera arvutamiseks järgnevat valemit:

$$I = E \times (D / (D + L \times r)) \times (D^2 / (D^2 + Q \times r^2))$$

Kus

- I on arvutatud valgusallika valgustugevus vastavas punktis.
- E on valgusjõu liuguri Energy väärtus.
- D on kauguse välja Distance väärtus.
- L on lineaarse liuguri Linear väärtus.
- Q on ruutsõltuvuse liuguri Quad väärtus.
- r on selle punkti kaugus valgusallikast.

Mõlema liuguri nullimine

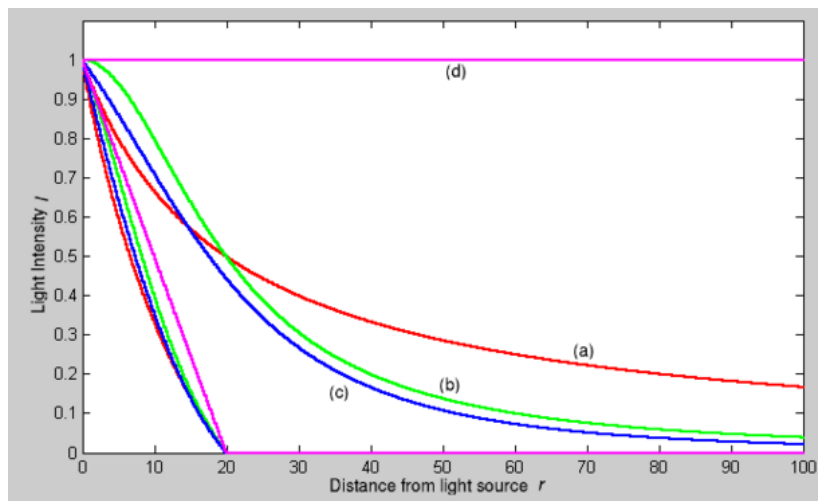
Kui mõlema liuguri (Linear ja Quadratic) väärtus on **0.0**, siis valgustugevus vahemaa kasvades ei nõrgene. See ei tähenda, et valgus enam üldse ei nõrgeneks (nõrgeneb küll), ent see juhtub ainult seetõttu, et valgusjõud hajub vahemaa kasvades suurema pinna peale. Hajunud valguse koguenergia jääb aga samaks mistahes kaugusel. Nähtava valguse hulka mõjutab ka valguse nurk. Niimoodi käitub valgus näiteks avakosmoses.

Juhul kui on vaja valgusallikat, mis ei nõrgene ja kiirgab sama palju valgustugevust igale pinnale oma teel, võib valgusti parameetrites kasutada konstantse väärtusega (Constant) nõrgenemiskõverat.

Nagu valemist võib lugeda, ei oma kauguse lahtris Distanceolev number mingit tähtsust, kui liugurite Linear ja Quad väärtused on **0.0**.

Kokkuvõtte pildis

Järgnev graafik võtab kokku lineaarse/ruutsõltuvusliku valguse nõrgenemise, kaasa arvatud variandid aktiivse kera nupuga Sphere (selgitatakse hiljem).



Valguse nõrgenemine:

a) Lineaarne (Linear=**1.0**, Quad=**0.0**); b) Ruutsõltuvuslik (Linear=**0.0**, Quad=**1.0**);

c) Lineaarne ja ruutsõltuvuslik (Linear=Quad=**0.5**); d) Null (Linear=Quad=**0.0**).

Lisaks on samade värvidega ära toodud samaväärsed kõverad aktiivse kera nupu Sphere korral.

Kasutaja määratud nõrgenemiskõver

Kasutaja määratud kõver (Custom Curve) on väga paindlik valguse nõrgenemiskõvera tüüp.

Ülejäänud nõrgenemiskõvera tüüpide puhul on valgusjõud alguses maksimaalne (otse valgusallika juures) ja langeb siis vahemaa kasvades etteantud valemi järgi.


Literal|Kasutaja määratud valguse nõrgenemiskõvera kasutamisel ilmub uus nõrgenemiskõvera paneel Falloff Curve.

Nõrgenemiskõvera graafik laseb muuta valguse intensiivsust erinevates nõrgenemiskõvera punktides (erinevatel kaugustel valgusallikast).

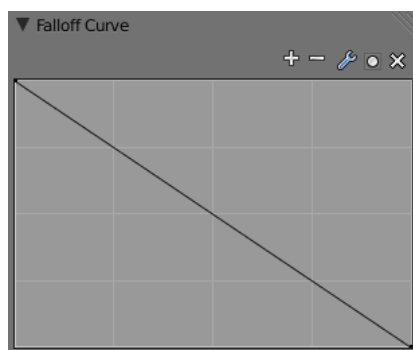
Nõrgenemiskõvera graafikul on kaks telge: "Kauguse" telg (*Distance*) ja "Valgustugevuse" telg (*Intensity*).

Kauguse telg (*Distance*) kujutab punkti asukohta valguskiire teel. Graafiku vasak äär vastab valgusallika asukohale ja parem äär punktile, kus allikast kiirgunud valgus on normaalselt täiesti kustunud. Me kasutame väljendit "normaalselt", sest nõrgenemiskõvera graafikut võib vajadusel sundida käituma täpselt vastupidi.

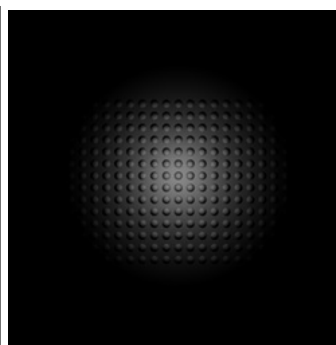
Valgustugevuse telg (*Intensity*) kujutab valguse tugevust teatud punktis valguskiire teel. Tugevamat valgust märgib kõrgem asend teljel, nõrgemat valgust madalam asend.

Nõrgenemiskõvera graafikut on lihtne muuta. Lihtsalt klõpsa LMB  graafikus punkti, mida soovid muuta, ning lohista see soovitud kohale. Klõpsates graafikus väikesele mustale ruudukujulisele käepidemele või selle lähedale, muutub ta valituks, mida tähistab värvi muutumine valgeks. Sa saad selle nüüd uude kohta tirida. Klõpsates graafikus kohale, kus käepidet veel ei ole, luuakse sinna uus käepide ning selle saab siis soovitud asukohta lohistada.

Järgnevas näitlikus graafikus (nõrgenemiskõvera Falloff Curve vaikesätted), algab valgus maksimaalse tugevusega valgusallika lähedal ning nõrgeneb lineaarselt paremale (valgustist eemaldudes).



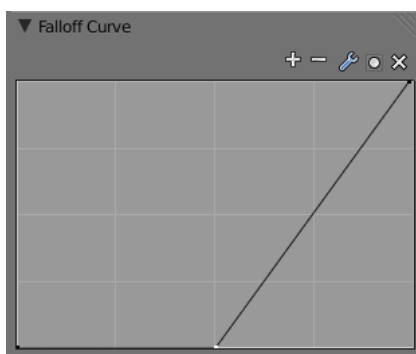
Nõrgenemiskõvera Falloff Curve paneeli vaikesätted.



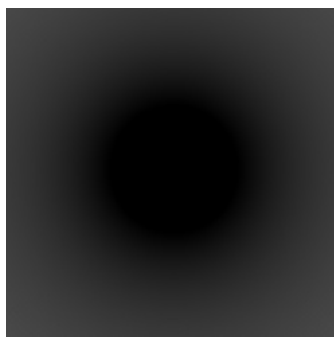
Renderdus kasutaja määratud nõrgenemiskõvera vaikesätetega.

Soovides nõrgenemiskõverat, kus valgus muutub valgustist kaugenedes intensiivsemaks, võib graafiku muuta selliseks:





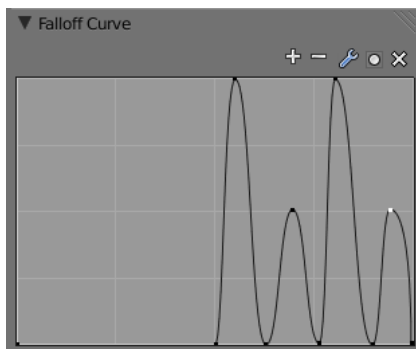
Ümberpööratud nõrgenemiskõvera graafik.



Renderdus ümberpööratud nõrgenemiskõveraga.

Ilmselgelt ei ole siinsed võimalused piiratud ainult lihtsate muutustega nagu graafiku ümber pööramine. Nõrgenemiskõverale võib anda mistahes soovitud kuju.

Siin on veel üks näide muudetud Literalnõrgenemiskõvera graafikust koos renderdustulemusega:

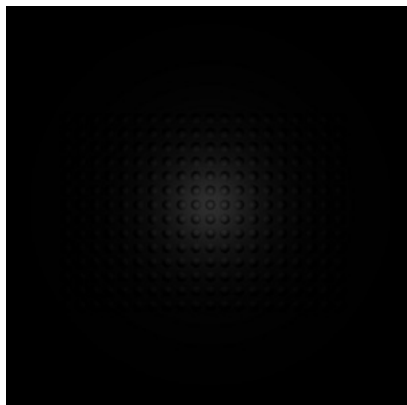


Võnkuv nõrgenemiskõver.



Võnkuv nõrgenemiskõver annab renderdatult lainetava efekti.

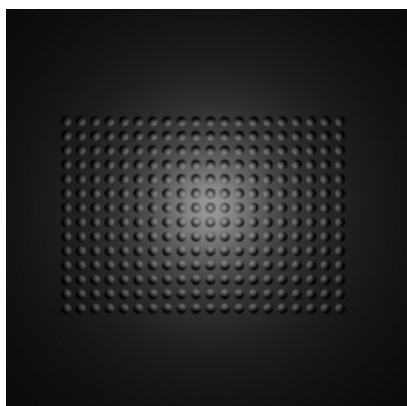
Pöördvõrdeline ruutsõltuvus (*Inverse Square*)



Renderdus Inverse Square nõrgenemiskõveraga, vaikesätted.

Seda tüüpi nõrgenemiskõveraga valgusti intensiivsus muutub pöördvõrdeliselt kauguse (Distance) ruuduga. See annab kõige kiirema ja realistlikuma nõrgenemise ja sobib kasutamiseks selliste valgustite nagu laua- ja tänavalampide simuleerimiseks. See on väga sarnane ruutsõltuvusliku (Quad) (ja vastavalt uue Lin/Quad Weighted nõrgenemiskõveraga, mille liugurite väärtused on Linear **0.0** ja Quad **1.0**).

Pöördvõrdeline lineaarne (*Inverse Linear*)

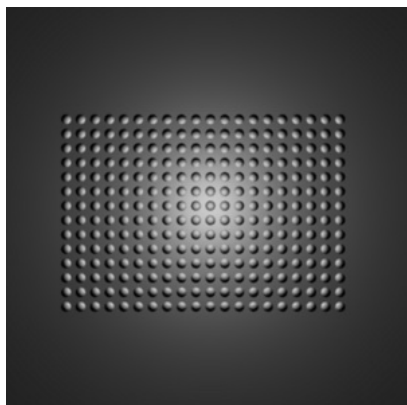


Renderdus Inverse Linear

nõrgenemiskõvera, vaikesätted.

Seda tüüpi nõrgenemiskõvera valgusti intensiivsus muutub pöördvõrdeliselt kaugusega Distance. See on kõigi valgustite vaikesäte, mis käitub samamoodi nagu vaikesäte varasemates Blenderi versioonides, kus ruutsõltuvuslikku vaikumiskõverat veel ei olnud – ja vastavalt nagu uus Lin/Quad Weighted kõver ligurite väärtustega Linear **1.0** ja Quad **0.0**. See kõver ei järgi täpselt füüsika reegleid, ent võib olla lihtsam kasutada.

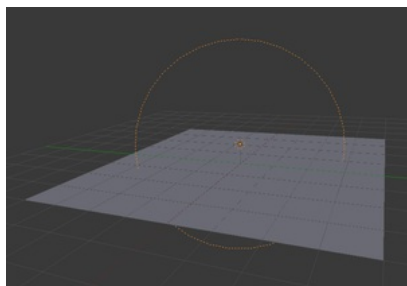
Ühtlane (*Constant*)



Renderdus Constant nõrgenemiskõvera, vaikesätted.

Seda tüüpi nõrgenemiskõvera valgusti intensiivsus kauguse kasvades ei muutu. See nõrgenemiskõver on kasulik kaugete valgusallikate (päike, taevast) simuleerimiseks, mis on nii kaugel, et nende valguse nõrgenemine ei ole märgatav. Päikesevalguse (Sun) ja poolkera (Hemi) valgustite nõrgenemiskõver on alati pidev.

Kera



Ekraanipilt 3D-vaatest, kus on näha kerajas valgust piirav ala (Sphere).

Kera (Sphere) piirab punkt- ja kohtvalgustite kiirguse leviku ulatust, takistades valguse levimist kaugemale lahtis Distance märgitud vahemaast.

Kui kera valik Sphere on aktiivne, ilmub valgusallika ümber punktiirjoon, mis märgib piiri, millest alates valgustugevus on null.

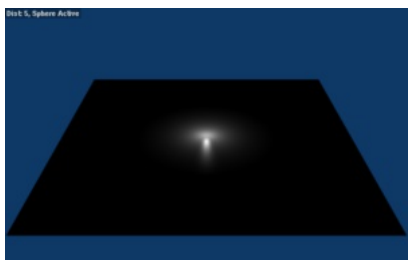
Kera valik Sphere lisab olemasolevale nõrgenemiskõvera valemile järgmise võrrandi:

$$I' = I \times (D - r) / D \text{ kui } r < D; \text{ muidu } 0$$

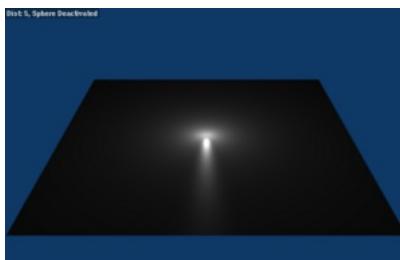
Kus:

- I' on lõplik valguse tugevus (kui valik Sphere on aktiivne).
- I on antud nõrgenemiskõvera järgi arvutatud valguse tugevus (ilma Sphere valikut).
- D on kauguse välja Distance välja väärtus.
- r on selle punkti kaugus valgusallikast.

Vaata joonist lineaarse/ruutsõltuvusliku (Lin/Quad Weighted) nõrgenemiskõvera lõigu lõpus.



Pidev (Constant) nõrgenemiskõver aktiivse kera valikuga Sphere.

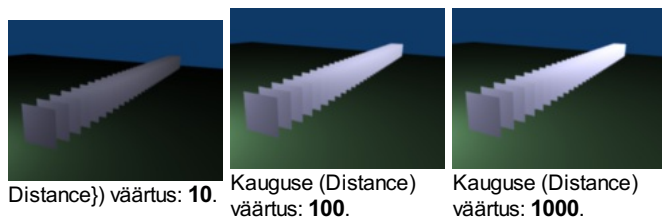


Pidev (Constant) nõrgenemiskõver ilma kera valikut Sphere.

Näidised

Kaugus

Selles näites on üks lamp asetatud tasapindade rivi lähedale. Seetõttu on valguse tugevus esimestel, keskmistel ja tagumistel tasapindadel oluliselt erinev. Vaadates "erinevaid kauguse (Distance) väärtusi", on näha, et suurendades kauguse valiku Dist väärtust, muutub üha enam objekte aina heledamaks.



Erinevad kauguse (Distance) väärtused (varjud keelatud).

Kauguse (Distance) väärtus määrab punkti, kus valgusjõud on langenud poole peale oma algsest väärtusest. Vaikesättena toimub valguse nõrgenemine lineaarselt. Selle väärtuse muutmise liigutab seda punkti, kus valgus on nõrgenenud poole võrra. Kaugust võib kujutada ümber valgusallika oleva kerana, mille igas punktis on valgusjõud täpselt pool sellest, mis ta oli valgusallika juures. Ei tohi unustada, et valgusjõud nõrgeneb edasi ka pärast valikuga kaugus määratud vahemaad. Kaugus määrab ainult punkti, kus pool valguse energiast on kadunud.

Joonisel (*Kauguse Distance väärtus: 1000*) paistavad ka kõige kaugemad objektid väga heledad. Kuna valguse oluline nõrgenemine on paigutatud niivõrd kaugemale, on valgus veel väga tugev isegi viimaste objektideni jõudes. Alles **1000** ühiku pärast on valguse tugevus langenud pooleni algsest tugevusest.

Joonisel (*Kauguse Distance väärtus: 10*) on kõige kaugemad objektid vaevalt valgustatud. Siin on valguse tugevus langenud poole võrra juba kümnenda objektini jõudes.

Võib tekkida küsimus, miks esimesed paar objekti paistavad teistest hämaramad, kuigi on valgustile kõige lähemal. See juhtub sellepärast, et valguse langemisnurk on nende objektide puhul lähedane täisnurgale tasapinna normaali suhtes. See on lambi tüüpi valgusti eripära. Kui selline valgusti viia lõpmatult kaugemale, hakkaksid ta omadused lähenema päikesele ehk päikesevalguse (Sun) tüüpi valgustile.

Pöördvõrdeline ruutsõltuvus

Inverse Square tüüpi nõrgenemiskõvera puhul nõrgeneb valgus mittelineaarselt, täpsemini pöördvõrdeliselt kauguse ruuduga. Inverse Square tüüpi nõrgenemiskõverale on iseloomulik, et algul kahaneb valgusjõud aeglaselt ning hiljem väga kiiresti.. Seda demonstreerivad pildid (*Inverse Square-tüüpi nõrgenemiskõver*).



Inverse Square-tüüpi nõrgenemiskõver (erinevate kauguse väärtustega).

Inverse Square-tüüpi nõrgenemiskõvera puhul määrab kauguse lahter (Distance) selle punkti, kus valguse nõrgenemine hakkab kiirenenema. Vaata ka kirjeldavat peatükki [valguse nõrgenemisest](#).

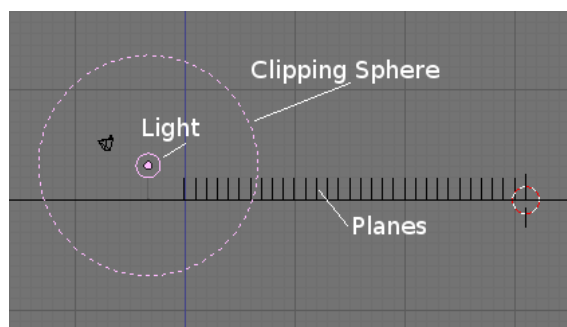
Näites (*Inverse Square tüüpi nõrgenemiskõver kauguse väärtusega 10*) langeb valguse tugevus nii kiirelt, et viimased objektid ei olegi üldse valgustatud.

Nii (*Inverse Square-tüüpi nõrgenemiskõver kauguse väärtusega 100*) kui ka (*Inverse Square-tüüpi nõrgenemiskõver kauguse väärtusega 1000*) näevad välja täpselt samasugused, sest nende kauguse väärtus Distance on määratud palju kaugemale viimasest objektist (mis on umbes 40 BU (Blenderi ühiku) kaugusel). Seetõttu on kõigile objektidele langev valgus peaaegu täie võimsusega.

Nagu eelnevalt selgitatud, paistavad ka siin esimesed paar objekti hämaramad, kuna paiknevad täpselt valgusallika all. Tuleb meele pidada, et objekti pinna heledus sõltub ka tema pinnanormaali ja valgusallikast tuleva valguskiire vahelisest nurgast.

See tähendab, et pinna heledus ei sõltu ainult valguse tugevusest, vaid ka tema langemise nurgast.

Kera



Valgust piirav kera.

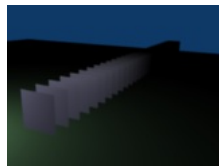
Valik kera (Sphere) muudab valguse intensiivsuse nulliks alates määratud kaugusest (Distance), sõltumata valitud nõrgenemiskõverast. Näites *Valgust piirav kera* on külgvaade aktiivse Sphere valikuga, kauguse väärtus on **10**.

Ühelegi objektile väljaspool seda kera ei lange sellest valgusallikast tulevat valgust.

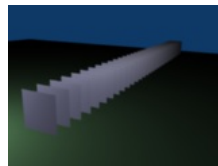
Selle valiku aktiveerimisega määrab kauguse väli Distance lisaks valguse nõrgenemiskõvera sätetele ka selle punkti, kus valguskiired täiesti ära nullitakse. Nb! Valguse intensiivsus ei kustu kera pinnal järsult, vaid sujuvalt (detaile vaata [kera kirjeldusest](#) ja [valguse nõrgenemiskõverate](#) nimekirjast).



Aktiivne kera valik
Sphere kauguse
väärtusega **10**.



Aktiivne kera valik
Sphere kauguse
väärtusega **20**.



Aktiivne kera valik
Sphere kauguse
väärtusega **40**.

Aktiivne kera valik Sphere erinevate kauguse sätetega, Inverse Linear-tüüpi valguse nõrgenemiskõver.

Näites (*Aktiivne kera valik Sphere kauguse väärtusega 10*) on kera raadius **10** ühikut, millest tulenevalt valgus kustub **10** ühiku kaugusel valgusallikast. Lineaarse nõrgenemiskõveraga on valgustugevus langenud suhteliselt madalale juba esimeste objektideni jõudes.

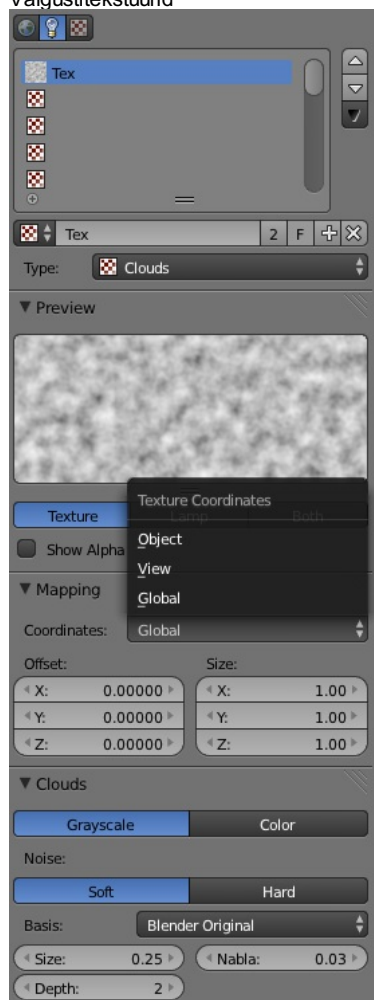
Näites (*Aktiivne kera valik Sphere kauguse väärtusega 20*) on kera raadius **20** ühikut ja valgus jõuab keskmiste objektideni.

Näites (*Aktiivne kera valik Sphere kauguse väärtusega 40*) on kera raadius **40** ühikut, mis on rohkem, kui vahemaa viimaste objektideni. Viimased objektid on siiski varjus, kuna valguse tugevus on sinna jõudes langenud nulli lähedale.

Näpunäited

Kui valgusallika sätetes on varjud keelatud, paistab ta ka läbi seinte ja muude sarnaste objektide. Kamina või küünlavalguse simuleerimiseks läbi akna nähtavas interjööris on kera valik Sphere väga käepärane. Täpse kauguse määramisega väljal Distance saad sa hoida sooja tulevalguse ainult ruumi sisemuses ning valgustada õue jaheda kuuvalgusega, kas siis päikesevalguse (Sun) või poolkera (Hemi) tüüpi valgusega (või mõlemaga).

Valgustitekstuuriid



Valgusti teksturi paneelid

Iga uus stseenile lisatud valgusti annab algul ühtlast, ühevärvilist valgust. Kuigi see võib lihtsate renderduste jaoks olla piisav, võimaldavad [valgustitekstuuriid](#) saavutada keerukamaid efekte. Vähemärgatava mustri tekstuurid võivad lisada valgusele nüansse, samas kui selge mustri tekstuurid saab kasutada selleks, et simuleerida tugevamaid efekte nagu diskokera, läbi puuvõra langevad päikeselaigud või isegi projektorid. Valgustitekstuuriid määratakse ühele kümnest kanalile. Nad käituvad täpselt nagu materjalitekstuuriid, kuid nad mõjutavad valguse värvust ja tugevust, mitte materjali pinna omadusi.

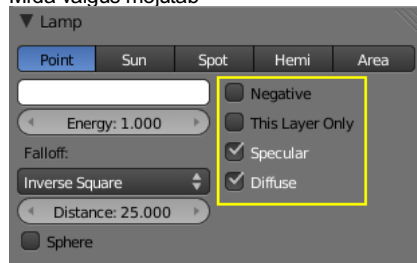
Valikud

Valgustitekstuuriid seaded on grupeeritud kahte paneeli. Siin peatükis vaatame ainult neid sätteid, mis erinevad tavaliste materjalitekstuuriid sätetest. Vaata peatükke [materjalide](#) ja [tekstuuriid](#) peatükke kõigile tekstuuridele ühiste seadete kohta.

Tekstuurispetsiifilised paneelid ja laotamise paneel Mapping on täpselt ühesugused. Siiski on valgustitekstuuriid märgatavalt vähem laotuse (Mapping) valikuid; kuna valgustil endal ei ole tekstuurikoordinaate, on võimalikud variandid ainult globaalne (Global), vaade (View) ja objekt (Object) ehk mõne teise objekti tekstuurikoordinaadid. Valgustitekstuuriid saab ka skaleerida ja nihutada.

Mõju paneelil Influence on samuti ainult väike osa tavalise materjalipaneeli valikutest. Valgustitekstuuriid saab seostada ainult valgusallika põhivärviga (Color) või varju värviga (Shadow). Kuna tekstuurid saab mõjutada ainult värve ja valgusallikal endal ei ole tekstuurikoordinaate, siis pole valikud põhivärv (Diffuse), läike värv (Specular), varjutus (Shading) ja geomeetria (Geometry) enam kättesaadavad.

Mida valgus mõjutab



Valgusti paneel Lamp, millel on valgust mõjutavad valikud välja on toodud

Igal valgustil on hulk valikuid, millega saab määrata, milliseid objekte ta valgustab ning kuidas ta mõjutab materjale.

Kiht (Layer)

Lubab valgustil valgustada ainult samal kihil olevaid objekte.

Negatiivne (Negative)

Valgusti poolt kiirguv valgus "lahutatakse" pinnale mujalt langevast valgusest, muutes pinna valgustamise asemel tumedamaks.

Hajusvalgus (Diffuse)

Ei lase lambil anda [hajusvalgust](#) (selline lamp reaalselt ei "valgusta" objekte).

Läikevalgus (Specular)

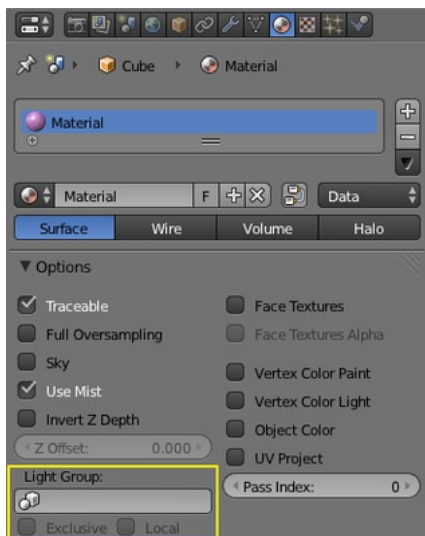
Ei lase lambil tekitada [läikeid](#).

Valgustitega seotud sätted

Need valikud on lähedaselt seotud valgustitega, kuid ei ole otseselt valgustite endi sätted.

Valgustigrupid

Materjalid

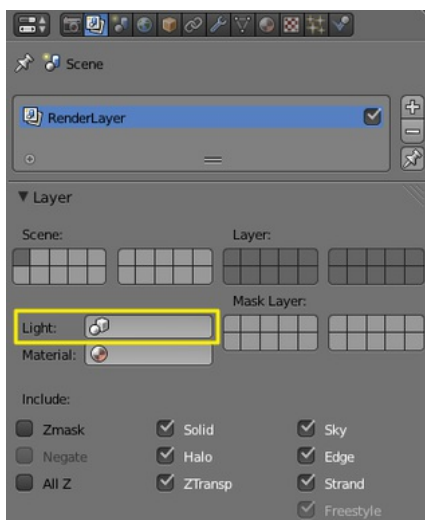


Valgustigrupi valik Light Group materjalide kontekstis Materials

Vaikesättena valgustavad kõik lambid kõiki materjale kõigil nähtavatel kihtidel, ent teatud materjalile (ja kõigile seda kasutavatele objektidele) võib määrata ainult teatud valgustite grupi. Selline seadistus võib olla kasulik keerulisema valgustusega stseenides. Seda saab sisse lülitada materjalide konteksti Materials seadete paneelis Options, valides väljal Light Group vastava lampide grupi. NB! Enne tuleb [valgustigrupp](#) luua.

Kui ainuvalik Exclusive on märgitud, mõjuvad valitud grupi valgustid *ainult* selle materjaliga objektidele.

Stseen



Valgustigrupi valik Light Group renderduskihtidele Render Layers

Sarnane valik on renderduskonteksti Render [renderduskihtide](#) paneelil Render Layers. Kui valguse lahtris Light määrata valgustigrupp, valgustavad seda stseeni ainult selle grupi valgustid.

Vaata ka

- [Valgustite sissejuhatus](#)
- [Varjud](#)
- [Materjalide sissejuhatus](#)

Varjud

Valgus ei eksisteeriks ilma oma vastandi ehk varjuta. Vari on objekti tumedam osa, kus valgus on osaliselt või täielikult blokeeritud ning ei valgusta objekti. Varjud lisavad stseenile kolmemõõtmelisust ning kontrasti. Reaalses maailmas ei ole praktiliselt ühtegi kohta, kus puuduksid varjud, seepärast on realistlike renderduste saamiseks varjude kasutamine väga oluline. Blender toetab järgnevaid varjutüüpe:

1. [Kiirtejälitusega valgusallikate varjud](#)
2. [Puhverdatud valgusallikate varjud](#)
3. [Kaudsed varjud \(*Ambient occlusion*\)](#)
4. [Kaudne valgustus \(*Indirect lighting*\)](#)

Kaudvari ei ole mitte valgusest, vaid objekti geomeetriast sõltuv vari. Ta on selles nimekirjas, kuna imiteerib seda, kui valgus ei saa objekti ühtlaselt ja täielikult valgustada. Kindlasti tuleb mainida ka üldvalgustust (Ambient lighting), kuna selle lisamine vähendab varjude mõju.

Erinevate tulemuste saamiseks võib kasutada kiirtejälituse meetodi ja puhverdatud varjude kombinatsiooni. Isegi kiirtejälituse meetodil heidetud varjude puhul jätavad erinevad valgustid erineva kuju ja tugevusega varje. Sõltuvalt valgustite paigutusest võib üks valgusti teise valgusti poolt loodud varju täiesti või osaliselt kaotada.

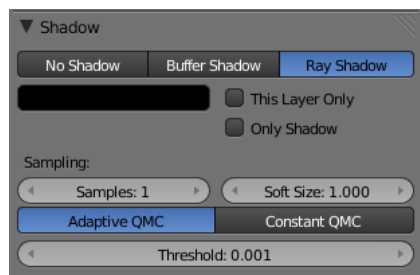
Varjude kasutamine nõuab mitmes eri kohas asuvate seadistuste koostööd:

1. Valgusti peab saama varje heita (varju sisse lülitamine ja valguse suund).
2. Läbipaistmatu objekt peab sattuma valguse teeale (asukoht ja kiht).
3. Teise objekti materjal peab olema võimeline varje vastu võtma (varju valik Shadow ja läbipaistvate vastuvõtmise valik Receive Transparent sisse lülitatud).
4. Renderdusmootor peab varjud arvutama (varju valik Shadow puhverdatud varjude jaoks ning varju valik Shadow ja kiire valik Ray kiirtejälitusega varjude jaoks).

Näiteks tavaline lamp (Lamp), pindvalgusti (Area) ja päikesevalgus (Sun) saavad heita kiirtejälitusega varje, kuid mitte puhverdatud varje. Kohtvalgusti (Spot) saab heita mõlemat tüüpi varje, poolkeravalgusti (Hemi) ei heida üldse varje. Kui päikesevalgus (Sun) on suunatud küljele, ei heida ta tasandist ülalpool asuvast poolkerast varje tasandile, sest valgus ei liigu selles suunas. Kõigil varje heitvatel valgustitel on mõningad ühised parameetrid, mida kirjeldatakse täpsemalt [siin](#).

Et valikuid oleks rohkem (ning asi segasem), saab valgustid ja materjalid panna vastavalt kas "ainult" varje heitma või vastu võtma ja mitte objekte valgustama ega nende värvi või läiget muutma. Samuti saavad renderduskihid varjude genereerimise sisse ja välja lülitada ja vastavalt võivad nende väljundid sisaldada või mitte sisaldada varjude andmeid.

Valgustid: kiirtejälituse meetodil (*Ray-traced*) heidetud varjud



Valgustil sisse lülitatud kiirtejälitusega varjud (Ray Shadow)

Kiirtejälitusega varjude heitmise meetod annab väga vähe mälu kasutades väga täpsed varjud, kuid seda suurema renderdusaja arvelt. Seda varjutüüpi saab kasutada kõikide valgusallikatega välja arvatud poolkera (Hemi).

Erinevalt [puhverdatud varjudest](#), genereeritakse kiirtejälitusega varjud, saates valgusallikast ühtlaselt kõigis suundades välja valguskiiri. Seejärel kiirtejälitaja märgib ära, milliseid piksleid pildil valguskiired tabasid ja milliseid mitte. Need pikslid, mida valguskiir ei taba, asuvad ilmselt varjus.

Iga valgusti heidab valguskiiri erineval viisil. Näiteks kohtvalgusti (Spot Light) heidab valgust ühtlaselt igas suunas koonusja valgusvihi sisemuses. Päikesevalgus (Sun) heidab valgust lõpmatult kaugel asetsevast punktist, kusjuures kõik valguskiired on paralleelsed päikese suunaga.

Kui kiirtejälitus on lubatud, siis suurendab iga stseenile lisatud valgusallikas renderdamiseks kuluvat aega. Kiirtejälituse meetodil genereeritud varjud nõuavad rohkem arvutamist kui puhverdatud varjud, ent tekitavad teravad servad, kasutades väga vähe operatiivmälu.

Kiirtejälitusega varjude sisselülitamiseks on vajalikud kolm tegevust:

- Lülitada varjud renderduskonteksti Render varjutuspaneelil Shading globaalselt sisse.
- Lülitada samas paneelis globaalselt sisse kiirtejälitus Ray tracing.
- Lülitada valgustil kiirtejälitusega varjud sisse, vajutades valgustikonteksti Light varjude paneelis Shadows kiirtejälitusega varjude nuppu (lülita Ray Shadow). Sõltuvalt valgustitüübist võib see paneel olla veidi erinev.
 - Kõigil kiirvarje heitvatel valgustitel on mõningad ühised parameetrid, mida kirjeldatakse täpsemalt peatükis [Kiirtejälitusega varjude parameetrid](#).

Kiirtejälitusega varje saavad heita järgmised valgusallikad:

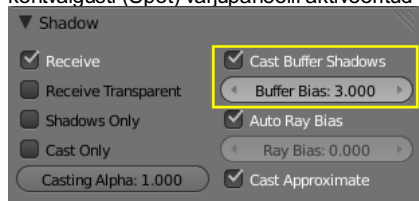
- [Punktvalgusti \(*Point*\)](#)
- [Kohtvalgusti \(*Spot*\)](#)
- [Pindvalgusti \(*Area*\)](#)

- [Päike \(Sun\)](#)

Valgustid: puhvervarjud



Puhverdatud varjud Buffer Shadow on kohtvalgusti (Spot) varjupaneelil aktiveeritud



Materjalil on puhverdatud varjude vastuvõtmine Cast Buffer Shadows sisse lülitatud

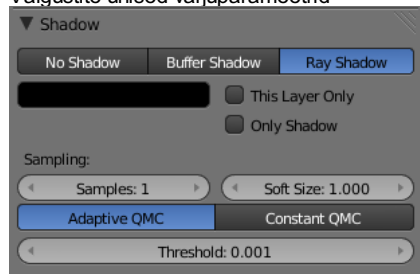
Puhverdatud varje renderdatakse kiiresti, ent seda täpsuse ja kvaliteedi arvelt. Puhverdatud varjud nõuavad ka rohkem arvuti operatiivmälu kui kiirtejälitusega varjud. See, kas puhverdatud varje kasutada, sõltub sinu vajadustest. Kui sa renderdad animatsiooni või kui sul ei ole aega keeruliste pehmete varjudega stseeni renderdamisega tunde oodata, on puhvervarjud hea valik.

Laotusjoone renderdaja - ja seda Blenderi sisseehitatud renderdaja muuhulgas ka on - saab arvutada varje varjupuhvri abil. See tähendab, et valgusti vaatepunktist (nagu oleks valgusti kaamera) "renderdatakse" stseeni pilt ja iga sellel pildil oleva punkti kohta salvestatakse selle kaugus kohtvalgustist. Kõik punktid tegelikult "renderdatud" pildil, mis on kohtvalgustist kaugemal, kui vastav punkt valgusti jaoks renderdatud pildil, loetakse varjus olevaks. Need andmed salvestatakse varjupuhvrisse.

Puhverdatud varjude kasutamiseks on vajalikud kaks tegevust:

- Lülita varjud globaalselt sisse, vajutades maailma konteksti World konteksti kogumispaneelis Gather nuppu Approximate.
- Lülita vastava valgusti varjud sisse, vajutades algustikonteksti Light varjupaneelil Shadow puhvervarjude nuppu Buffer Shadow.
- Veendu, et iga materjali jaoks on materjalikonteksti Material varjupaneelis Shadow puhvervarjude heitmise valik Cast Buffer Shadows sisse lülitatud.
- Ainus valgustitüüp, mis saab puhvervarje heita, on [kohtvalgusti](#).

Valgustite ühised varjuparameetrid



Valgustite ühised varjuparameetrid

Kõigil valgustitel, mis saavad heita varje ([lamp \(Lamp\)](#), [kohtvalgusti \(Spot\)](#), [pindvalgusti \(Area\)](#) ja [päikesevalgus \(Sun\)](#)) on mõned ühised seadistused, mida siin kirjeldame:

Sama kihi objektid (This Layer Only)

Kui see on sisse lülitatud, siis valgustatakse ainult neid objekte, mis on valgustiga samal kihil.

Ainult varjud (Only Shadow)

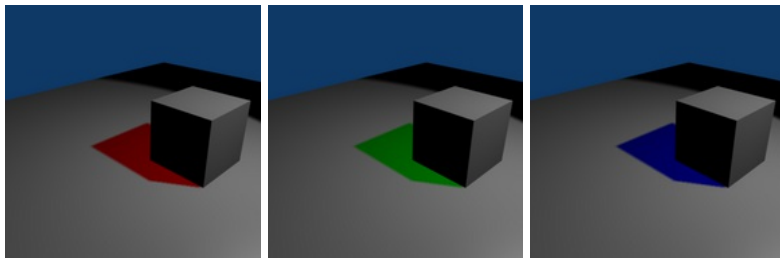
Objekti ei valgustata, ent luuakse varjud, mis tekiksid objekti valgustamise puhul.

Seda meetodit kasutatakse tihti selleks, et määrata, kuhu ja kuidas varjud langevad. Objekti valgustatakse valgusallikaga, mis ei heida varje, ja lisatakse hoopis teistsugune valgusallikas, mis ei valgusta, aga millel on ainult varju valik Only Shadow aktiivne. Viimast nihutades saab varjude asukohta täpselt määrata.

Varjude värvus

Selle värvivalijaga saab valida heidetavate varjude värvust (vaikimisi on need mustad).

Alltoodud pildid on renderdatud valge valgusega, varieerides varjude värvust.



Näidis punast värvi varjust

Näidis rohelist värvi varjust

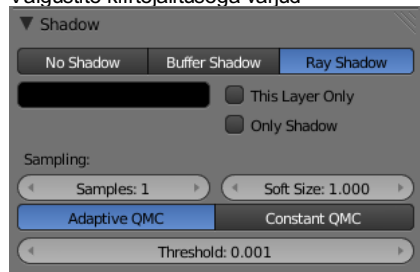
Näidis sinist värvi varjust

Kuigi varjude värviks saab määrata ka valge, kaotab see varjud lihtsalt ära.

Vaata ka

- [Varjud](#)
- [Üldised kiirtejälituse parameetrid](#)
- [Lambi kiirtejälitusega varjud](#)
- [Kohtvalgusti kiirtejälitusega varjud](#)
- [Pindvalgusti kiirtejälitusega varjud](#)
- [Päikesevalguse kiirtejälitusega varjud](#)
- [Kohtvalgusti puhverdatud varjud](#)

Valgustite kiirtejälitusega varjud



Valgustite kiirtejälitusega varjude parameetrid

Enamikul valgustitüüpidest ([lamp \(Lamp\)](#), [kohtvalgusti \(Spot\)](#) ja [päikesevalgus \(Sun\)](#)) on ühised kiirtejälitusega varjude parameetrid. NB! Kuigi [pindvalgus \(Area\)](#) kasutab põhiliselt samu valikuid, on tal mõned eripärad, mida on täpsemalt kirjeldatud [pindvalguse kiirtejälitusega varjude](#) peatükis.

Kiirtejälitusega varjud (Ray Shadow)

Kiirtejälitusega varjude valik Ray Shadow lubab valgusallikal heita varje kiirtejälituse meetodil.

Kui Ray Shadow valik on aktiivne, ilmub täiendav valikute komplekt:

Kiirtegeneraatori tüüp

Kiirte asendi genereerimismeetod varjude arvutamisel: Adaptive QMC on kõige kiirem, Constant QMC tekitab vähem müra, kuid on aeglasem. Sellega saab valida, millist algoritmi kasutatakse kiirtejälitusega varjude testkiirte genereerimiseks (hetkel on võimalik kasutada kahte kvaasi-Monte-Carlo (QMC) varianti; loe nende kohta lähemalt [siit](#)):

Ühtlane QMC (Constant QMC)

Constant QMC paigutab varjude testkiiri väga ühtlaselt ja samase tihedusega. See meetod annab väga häid arvutustulemusi ja on palju täpsem, ent oluliselt aeglasem kui adaptiivne meetod (Adaptive QMC).

Adaptiivne QMC (Adaptive QMC)

Adaptiivse QMC meetodiga (Adaptive QMC) paigutatakse testkiired veidi ebaühtlasemal viisil. See meetod annab suhteliselt häid tulemusi, ent siiski mitte nii häid kui ühtlane QMC (Constant QMC). Adaptiivse QMC (Adaptive QMC) eelis on see, et ta on üldjuhul oluliselt kiirem ning mitte oluliselt halvem kui ühtlane QMC (Constant QMC).

Testkiired (Samples)

Kasutatavate testkiirte arv (tegelik arv on selle väärtuse ruut).

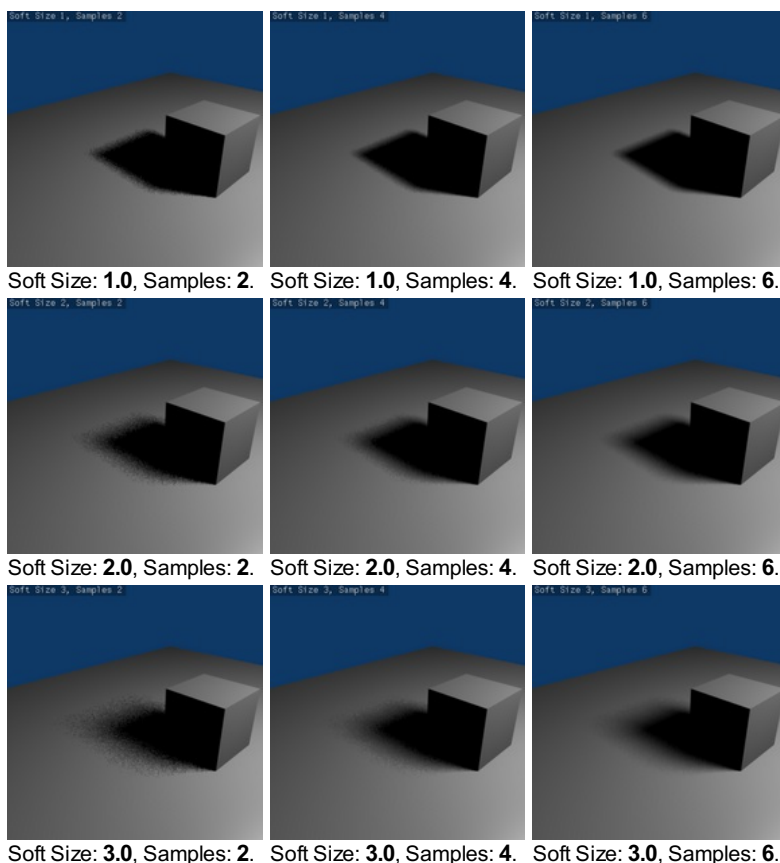
See liugur määrab maksimaalse testkiirte arvu, mida kasutatakse nii ühtlase (Constant) kui ka adaptiivse (Adaptive) QMC varjude arvutustes. Liuguri maksimaalne väärtus on **16** – tegelik testkiirte arv on selle numbri ruut; näiteks väärtus **3** tähendab, et tegelikult kasutati $3^2 = 9$ testkiirt.

Pehmus (Soft Size)

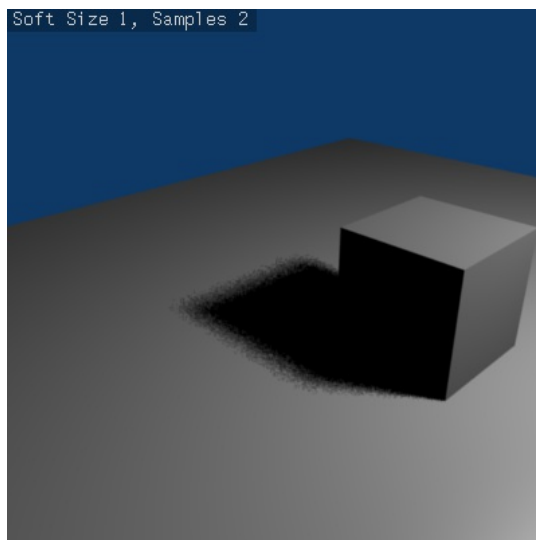
Valgusallika suurus kiirtejälitusega varjude arvutamiseks. See liugur määrab hajusa/pehme poolvarju suuruse varju äärel.

Pehmus Soft Size määrab ainult poolvarju laiuse, mitte aga selle, kui ühtlane ja sujuv on üleminek. Kui soovid laia poolvarju, mis on sealjuures sujuva ja ühtlase üleminekuga, tuleb ka testkiirte lahtirisse (Samples) sisestada suurem väärtus kui **1**, vastasel korral pole sellel väljal mingit mõju ja varjude äär ei ole hajus. Maksimaalne pehmuse (Soft Size) väärtus on **100.0**.

Järgnev tabel demonstreerib nende seadistuste mõju varju äärtele, näidates renderdusi erinevate pehmuse (Soft Size) ja testkiirte (Samples) sätetega:



Allpool on toodud animeeritud versioon tabelis toodud piltidest, mis toob välja efektide erinevused:



Animatsiooni nägemiseks pead võib-olla pildile klõpsama.

Lävi (Threshold)

Adaptiivsete testkiirte lävi. Seda lahtrit kasutatakse varjude arvutamisel adaptiivse QMC (Adaptive QMC) meetodil. See väärtus määrab valgustatuse taseme, millest allpool adaptiivne QMC võib kiirte genereerimise lõpetada. Läve välja Threshold maksimaalne väärtus on **1.0**.

Mis on kvaasi-Monte Carlo?

Monte Carlo meetod on meetod, kus teatud alalt või selle lähedusest kogutakse juhuslikult katseandmeid (mistahes andmeid, näiteks valguse, värvi või peegelduste kohta) ja nende põhjal hinnatakse teatavate arvutuste, mis tavaliselt vajavad mitut andmepunkti täpsuse hindamiseks, ligilähedane väärtus. Monte Carlo meetod üritab olla võimalikult juhuslik. Selle tulemusena võib analüüsitud aladel esineda suuri ebaregulaarseid lünki (kohti, kust andmeid pole kogutud) ja see omakorda võib tekitada probleeme teatud arvutuste (näiteks varjude arvutamise) korral.

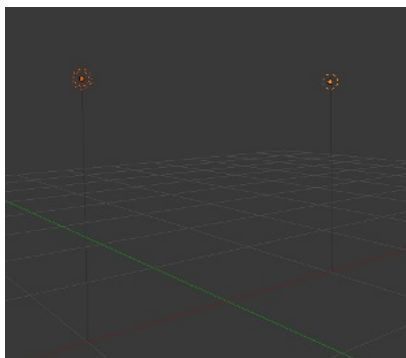
Selle probleemi lahenduseks on kvaasi-Monte Carlo meetod.

Kvaasi-Monte Carlo meetod on samuti juhuslik, kuid üritab tagada seda, et katseandmete kogumispunktid on paremini jaotatud (jättes vähem korrapäratuid lünki) ja katavad katseala võimalikult ühtlaselt. Seetõttu annab see meetod mõnikord paljude andmepunktide järgi tehtud arvutustes täpsemaid tulemusi.

Valgustid

Blenderis on viis erinevat valgustitüüpi, millest igaühel on omad eripärad ja tugevad ning nõrgad küljed. Siin on nimekiri kõigist valgustitüüpidest:

- [Lamp](#) (Lamp) on kõigis suundades ühesugune punktvalgus, väga sarnane lambipimile.
- [Kohtvalgusti](#) (Spot) on suunatud punktvalgus, mis sarnaneb kohtvalgustile.
- [Pindvalgus](#) (Area) simuleerib helendavat pinda nagu näiteks aknad, neoontahvlid, televiisorekraanid.
- [Poolkera](#) (Hemi) simuleerib väga suure pinnaga ja kaugel asuvat valgusallikat, nagu näiteks taevas.
- [Päikesevalgus](#) (Sun) simuleerib väga kaugel asuvat punktvalgustit, näiteks päikest.



Kahe punktvalgusti kõrguse ja varju tähistused. Vasakpoolse lambi puhul on kiirtejälitusega varjud sisse lülitatud.

Stseeni saab lisada uusi valgusteid lisamise menüüst Add ülemisel päisel või vajutades ⇧ ShiftA » Add » Lamp (lisa -> lamp).

Kui lamp on stseeni lisatud, paistab tema asukoht 3D-vaates ringist ümbritsetud punktina, mõne lambitüübi puhul on näha ka punktiirjoonega võred, mis täpsustavad valgusti parameetreid ja suunda. Igal valgustitüübil on omad tähis, ent mõned näidikud on ühised kõigile valgustitele:

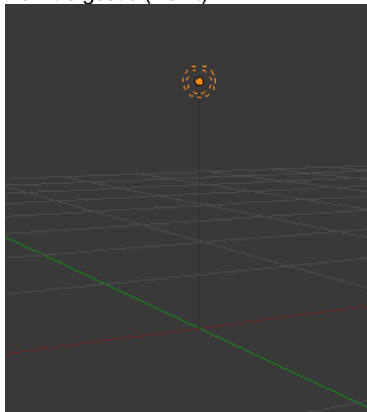
Varjud (Shadows)

Kui varjud on sisse lülitatud, lisandub valgusti tingmärgi pidevringi ümber punktiirjoonega ring. Selle järgi on lihtne ja kiire aru saada, kas valgusti jaoks on varjud sisse lülitatud või mitte.

Vertikaalne kõrgusemärgis (Vertical Height Marker)

See tuhm hall joon aitab määrata valgusti asukohta X-Y tasandi suhtes.

Punktvalgustid (*Point*)



Punktvalgusti

Punktvalgusti (Point) on ilma suunata valguspunkt: see tähendab, et ta kiirgab igase suunda sama palju valgust. Selle tingmäärgiks on ringiga ümbritsetud punkt. Kuna punktvalgusel ei ole suunda, määratleb objektile langeva valguse suuna valgustit ja objekti pinda ühendav joon.

Punktvalgusti valgustugevus/valgusjõud kahaneb (muuhulgas) sõltuvalt tema kaugusest objektist. Teisisõnu on kaugemal olevad pinnad renderdusel tumedamad.

Valgusti parameetrid

Üldised parameetrid

Vahemaa (Distance), valgusjõud (Energy) ja värvus (Color)

Need sätted on kõigil valgustitüüpidel samad ning on lähemalt kirjeldatud peatükis [Valgusti parameetrid](#).

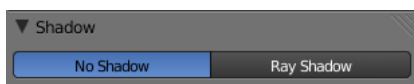
Negatiivne (Negative), ainult sellele kihile (This Layer Only), läige (Specular) ja hajusus (Diffuse)

Need sätted määravad selle, kuidas valitud valgusti objektile mõjutab, lähemalt selgitatud peatükis [Mida valgus mõjutab](#).

Vaibumine (Falloff) ja kera (Sphere)

Need sätted määravad selle, kuidas valgustist kiirgav valgus oma teekonnal nõrgeneb. Täpsemalt loe peatükist [Valguse nõrgenemine](#).

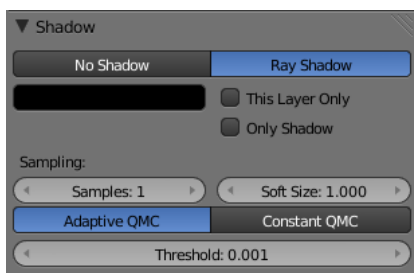
Varjud



Ilma kiirejälitusvarjudeta

Kui valgusallikaks on valitud punktvalgusti, on varjude paneelil Shadow järgmised vaikesätted:

Varjud kiirtejälituse meetodil



Punktvalgusti kiirtejälitusvarjudega ja sisse lülitatud adaptiivse QMC (Adaptive QMC) testiirte generaatoriga

Punktvalgus saab varje heita ainult kiirtejälituse meetodiga. Põhilised varjudega seotud valikud on samad kui teistel lambitüüpidel ning on täpsemalt kirjeldatud peatükis [Varjude parameetrid](#).

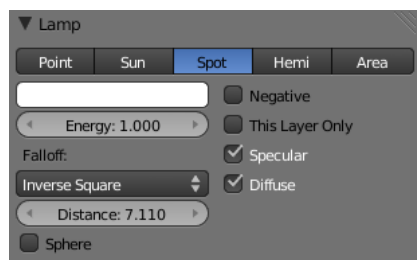
Kiirtejälituse meetodil heidetud varjudega seotud valikud on samad mis teistel lambitüüpidel ja neid on täpsemalt kirjeldatud peatükis [Kiirtejälituse parameetrid](#).

Kohtvalgusti (Spot)

Kohtvalgusti (Spot) heidab koonusekujulise valgusvihi etteantud suunas.

Kohtvalgusti on valgusallikatest kõige keerulisem ja oli pikka aega ka kõige laiemalt kasutusel, sest see oli ainus valgusti, mis sai varje heita. Kuna praeguseks on Blenderi renderdusmootoris lisatud kiirtejälitaja, saavad kõik valgustid (välja arvatud poolkera (Hemi)) varje heita. Sellest hoolimata on kohtvalgustite varjupuhvrid palju kiiremad renderdada kui kiirtejälituse meetodil heidetud varjud (eriti kui soovitakse hägusaid või pehmeid varje) ja neil on palju teisi võimalusi, nagu näiteks kolmemõõtmelised halod.

Valgusti parameetrid



Tavalised kohtvalgusti parameetrid

Üldised parameetrid

Kaugus (Distance), valgusjõud (Energy) ja värvus (Color)

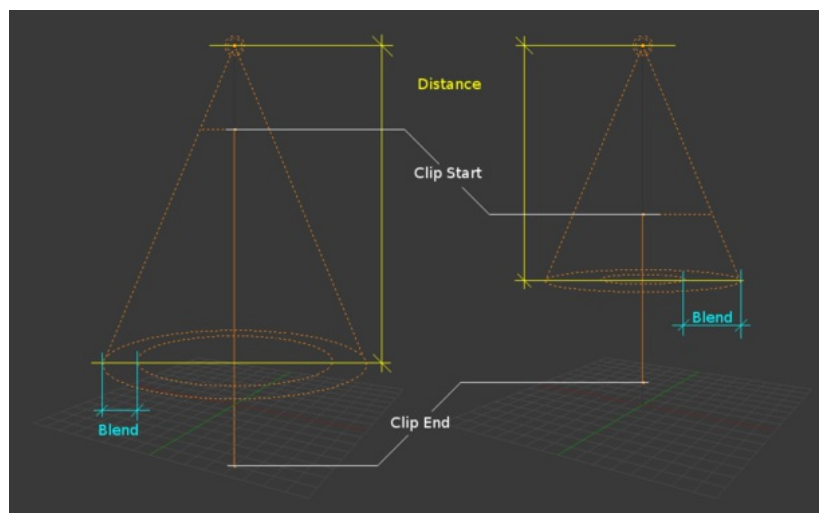
Need sätted on kõigil valgustiüüpidel samad ning neid on lähemalt kirjeldatud peatükis [Valgusti parameetrid](#).

Ainult sellel kihil (This Layer Only), negatiivne (Negative), hajusus (Diffuse) ja läige (Specular)

Need sätted määravad, milliseid objekti omadusi valgusallikas mõjutab, nagu kirjeldatud peatükis [Mida valgus mõjutab](#).

Vaibumine (Light Falloff) ja kera (Sphere)

Need sätted määravad selle, kuidas valgustist kiirgav valgus oma teekonnal nõrgeneb. Täpsemalt loe peatükist [Valguse nõrgenemine](#).



Kohtvalgusti parameetrite muutmine muudab ka selle välimust 3D-vaates.

Varjude parameetrid

Vari puudub (No Shadow)

lubab ja keelab sellel valgustil varjude heitmist. See võib olla kasulik, kui stseenile on vaja lisada tagasihoidlikku suunatud valgust.

Puhverdatud varjud (Buffer Shadow)

Loe peatükki [Kohtvalgusti puhverdatud varjud](#)

Kiirte varjud (Ray Shadow)

Loe peatükki [Kohtvalgusti kiirtejälitusega varjud](#)

Kohtvalgusti võib heita varje kas kiirtejälituse meetodil või [varjupuhvri abil](#). Mõlemal varjutüübil on erinevad lisavõimalused.

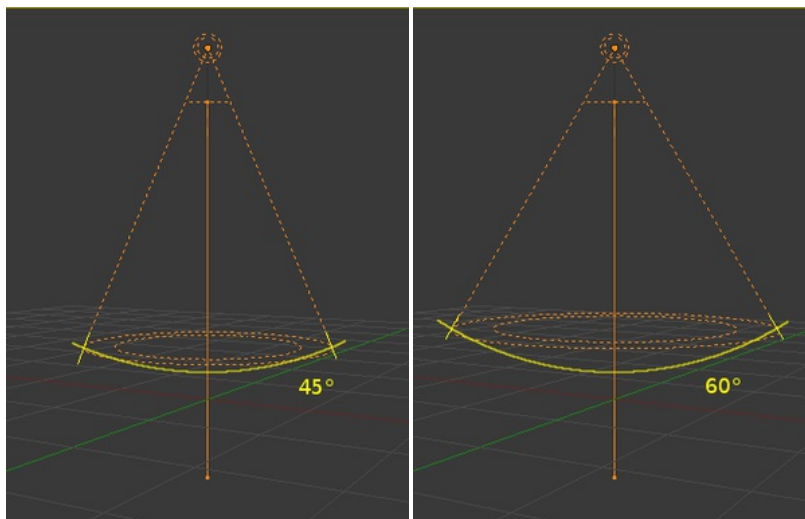
Kiirtejälitusega varjud on tavaliselt täpsemad ja neil on täiendavad lisavõimalused nagu läbipaistvad varjud, ent nad teevad renderdamise aeglaseks.

Puhverdatud varjude tekitamine on palju keerulisem ning nad sisaldavad olulisi lihtsustusi, kuid nende renderdamine läheb palju kiiremini. Osad varjudega seotud valikud on siinkohal samad kui teistel lambitüüpidel ning on täpsemalt kirjeldatud peatükis [Varjude parameetrid](#).

Valgusvihi parameetrid

Suurus (Size)

Kohtvalgusti valguskoonuse välimine nurk, mis määrab põhiosas kohtvalgusti poolt tekitatava valguslaigu suuruse. Liugur muudab valguskoonuse tipunurga suurust, võimalikud väärtused on **1.0°** kuni **180.0°**.



Valgusvihi suuruse (Size) muutmine

Sulanda (Blend)

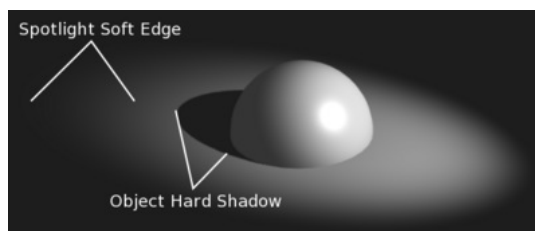
Sulandamise liugur Blend määrab kohtvalgusti sisemise koonuse suuruse. Blend liuguri väärtus võib olla **0.0** kuni **1.0**. See väärtus on suhteline ja näitab, kui suure osa välimise koonuse suurusest (Size) peaks sisemine koonus enda alla võtma.

Sisemise koonuse piir määrab nurga, millest väljapoole hakkab kohtvalgusti valgus hajuma/pehmenema. Sisemise koonuse sees on valgus maksimaalse tugevusega. Mida suurem Blend väärtus, seda hajusam/pehmemate äärtega on valgusvoog ning seda väiksem on sisemise koonuse poolt heidetud täies valgusjõus valguslaik (kuna hajumine algab varem).

Järsema hajumiskõveraga valguse ja vähem hajutatud servadega valguslaigu saamiseks tuleb väljal Blend olevat väärtust kahandada. Blend väärtus **0.0** annab kohtvalgustele väga teravad servad ilma mingi üleminekuta valguse ja varju vahel.

Kohtvalgusti poolt heidetav valgusvihi nõrgenemine külgsuunas on määratud valikute Blend ja Size väärtustega. Mida suurem on nende kahe väärtusega määratud ringikujuline ala, seda sujuvamalt valgus külgsuunas nõrgeneb.

Valikud Blend ja Size mõjutavad ainult kohtvalgusti valgusvihi laiust ja pehmust (radiaalset valguse nõrgenemist), mitte varju pehmust, nagu näha juuresoleval pildil.



Renderdus, kus on näha pehme äärega valguslaik ja terava äärega vari.

Nagu näha ülaltoodud pildil, on objekti vari tänu kiirtejälitusele terav, ent valguslaigu serv on hajus. Kui sa tahad, et objektide varjud oleksid kohtvalgusti valgusvihus hajusate äärtega, tuleb muuta varjude sätteid.

Ruut (Square)

Ruudu valik Square muudab kohtvalgusti poolt heidetava valguslaigu ruudukujuliseks. Vaikesättena on valguslaik ringikujuline.

Näita koonust (Show Cone)

Valik Show Cone joonistab 3D-vaates läbipaistva koonuse, mis näitab, millised objektid jäävad valguse sisse.

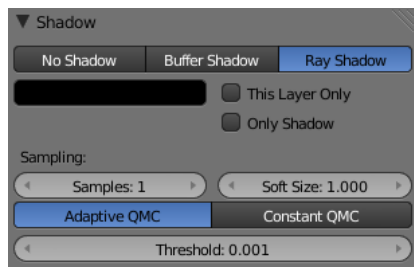
Halo

Valik Halo lisab kohtvalgustile ruumilise helenduse. Vaata ka [Kohtvalgusti halod](#).

Puhverdatud vari

Puhverdatud varje (Buffered Shadows) nimetatakse ka sügavuskaardi (*depth map*) varjudeks. Varjud arvutatakse objektide kauguse põhjal valgusallikast. Rohkem infot puhverdatud varjude kohta leiab peatükist [Puhverdatud varjud](#).

Kohtvalgusti varjud kiirtejälituse meetodil



Varjude paneel Shadow, mille tüübiks on valitud kiirte varjud (Ray Shadow).

Kiirtejalituse meetodil heidetud varjudega seotud valikud on samad mis teistel lambitüüpidel ja neid on täpsemalt kirjeldatud peatükis [Kiirtejalituse parameetrid](#).

Kohtvalgusti ruumilised efektid (*volumetric effects*)



Kohtvalgusti Halo valikud

Kohtvalgusti võib tekitada ka ruumilisi efekte. Täpsemalt loe peatükist [Valgusvihud](#).

Halo

Valik Halo võimaldab lisada kohtvalgustile ruumilisi efekte. Et valgusvihud oleksid nähtavad, peab nupp Halo olema sisse lülitatud. Puhverdatud varje kasutades ilmuvad lisavalikud, nagu kirjeldatud peatükis [Kohtvalgusti puhverdatud varjud](#).

Intensiivsus (Intensity)

Liugur Intensity määrab valgusti poolt heidetavate valgusvihkude intensiivsuse/tiheduse. Mida madalam on intensiivsuse väärtus, seda vähem märgatavad on valgusvihud. Mida suurem on intensiivsus, seda selgemad ja tihedamad on valgusvihud.

Aste (Step)

Selle lahtri väärtus võib olla **0** kuni **12**. See määrab, kas ja millise kvaliteediga valgusvihke see kohtvalgusti heidab.

Kui astme (Step) väärtus on **0**, siis valgusvihke ei tekitata.

Erinevalt enamikest teistest valikutest, kui astme väärtus kasvab, siis valgusvihkude kvaliteet kahaneb (kuid nende renderdamine võtab vähem aega) ning vastupidi.



Valiku Step väärtused

Halo Step väärtus **8** on enamasti sobiv kompromiss kiiruse ja täpsuse vahel.

Blender simuleerib kohtvalgustite valgusvihke ainult oma sisemist renderdusmootorit kasutades. Valgusti valgusjõu (Energy) ja halo intensiivsuse (Intensity) sätteid muutes võib saavutada väga kummalisi tulemusi.

Näiteks väga madala või olematu valgusjõuga (Energy) kohtvalgusti annab väga intensiivse halo puhul tumeda nimbuse, mis reaalses maailmas ei oleks võimalik. Kui Blenderi sisemise renderdusmootoriga halo kasutada, tuleb seda silmas pida.

Märkus

Haloefekti saab oluliselt võimendada puhverdatud varje kasutades: kui halo astme (Step) väärtus erineb nullist, tekivad ruumilised varjud (varjude vihud). Rohkem infot puhverdatud varjude kohta leiab peatükist [Puhverdatud varjud](#).

Kohtvalgusti puhverdatud varjud



Kohtvalgusti (Spot) sisselülitatud puhverdatud varjudega (Buffer Shadow)

Kohtvalgustid võivad kasutada kas [kiirtejälitusega varje](#) või puhverdatud varje. Mõlemal varjutüübil on erinevad lisavõimalused.

Kiirtejälitusega varjud on tavaliselt täpsemad ja neil on täiendavad lisavõimalused nagu läbipaistvad varjud, ent nad teevad renderdamise aeglaseks.

Puhverdatud varjude tekitamine on palju keerulisem ning nad sisaldavad olulisi lihtsustusi, kuid nende renderdamine läheb palju kiiremini. Osad varjudega seotud valikud on siinkohal samad kui teistel lambitüüpidel ning on täpsemalt kirjeldatud peatükis [Varjude parameetrid](#).

Varjupuhvri tüübid

Kui puhverdatud varjud (Buffer Shadow) on sisse lülitatud, heidab valitud kohtvalgusti (Spot) varje varjupuhvri, mitte kiirtejälituse abil. Varjude paneelile Shadow ilmuvad sellega mitmed lisavalikud.

Puhvri tüüp (Buffer type)

Puhverdatud varjude loomiseks on mitu viisi. See valik määrab, millist varjupuhvri generaatorit kasutada.

Kokku on neli erinevat varjude genereerimise viisi:

- Klassikaline (Classical)
- Klassikaline-keskmistatud (Classic-Halfway)
- Ebaregulaarne (Irregular)
- Sügav (Deep)

Lisainformatsiooni erinevate varjude genereerimise meetodite kohta leiab järgmiste linkide alt:

- [Development Release Logs 2.43: Irregular Shadow Buffer](#)
- [Blender Nation: Blender Gets Irregular Shadow Buffers](#)
- [Development Release Logs 2.43: Shadow Buffer Halfway Average](#)

“Klassikaline” ja ”keskmistatud klassikaline” varjude genereerimine



Puhverdatud varju tüübiks on määratud klassikaline-keskmistatud (Classic-Halfway)

Klassikaline (Classical)

Blenderi vanad vaikesätted kasutasid seda varjude genereerimise meetodit. Klassikaline meetod on vana puhverdatud varjude genereerimine, ent sellel oli tihti peale probleeme varjude täpsusega ja ta on väga tundlik varjupuhvri lahutuse (Shadow Buffer→Size (varjupuhver -> suurus)) ja erinevate nihke (Bias) väärtuste suhtes. Lisaks tekitab see generaator tihti peale artefakte objekti enesevarjutamisel.

Klassikaline varjude genereerimise meetod on iganenud ning seda toetatakse ainult vanade Blenderi versioonidega ühilduvuse tagamiseks. Kõigil muudel juhtudel tuleks kasutada meetodit klassikaline-keskmistatud (Classic-Halfway).

Klassikaline-keskmistatud (Classic-Halfway)

See täiustatud varjugeneraator on Blenderi vaikesäte varjudepuhvri loomisel. Selle algoritmi põhimõte on võtta lähima ja järgmise Z-sügavuse väärtuse keskmine, mis võimaldab nihke (Bias) väärtust vähendada ilma olulisi enesevarjutuse probleeme

tekitamata.

Nihke (Bias) väärtuse hoiuline väiksenä aitab kaasa varjude täpsusele, kuna suured nihke väärtused võivad põhjustada varjude kadumist väiksematel külgedel ning üldist varjude nihkumist.

Keskmiastatud klassikaline varjugeneraator ei tööta väga hästi omavahel kattuvate külgede puhul ning võib sel puhul ikkagi põhjustada nihkeprobleeme.

Sellel varjude genereerimise algoritmil on järgmised seadistused:

Suurus (Size)

Suurus (Size) võib omada väärtust **512** kuni **10240**. See määrab loodava varjukaardi lahutuse. Varjukaart määrab ära stseenis olevate varjude paiknemise.

Näiteks kui välja Size väärtuseks on **1024**, kirjutatakse valitud valgusti heidetavate varjude andmed **1024×1024** suuruse lahutusega ruudukujulisse puhvrisse.

Mida suurem on suuruse (Size) väärtus, seda parema eraldusvõime ning täpsusega heidetakse varjud, eeldusel et ülejäänud valguse ja stseeni parameetrid jäävad samaks. See kasutab aga rohkem mälu ja protsessori tööaega. Kehtib ka vastupidine: mida väiksem on suuruse (Size) väärtus, seda madalama kvaliteediga on varjud, kuid mälu ja protsessori tööaja vajadus on väiksem.

Lisaks valiku Size väärtusele mõjutab kohtvalgusti puhverdatud varju kvaliteeti ka valgusti valguskoonuse välisnurga suurus (valguslaigu kuju paneeli Spot Shape suuruse lahter Size).

Kui valguslaigu suurus (Size) kasvab, kahaneb heidetud varjude kvaliteet. Selle põhjus on see, et kui kohtvalgusti valgustatud ala suurust tõsta (suurendades valguskoonuse nurka (Size)), venitatakse varjupuhvri pindala suuremaks, et see kataks kogu valgustatud ala.

Valguskoonuse nurga (Size) suurendamine ei muuda varjupuhvri eraldusvõimet ja seetõttu varjude kvaliteet kannatab. Kui sa tahad, et varjude kvaliteet jääks samaks, tuleb valguskoonuse laiust (Size) suurendades tõsta ka varjupuhvri suurust (Size).

Eelnevat võib lühidalt kokku võtta nii:

Varjude kvaliteedi säilitamiseks laia nurgaga kohtvalgusti puhul on vaja tõsta varjupuhvri suurust (Size). Kehtib ka vastupidine: varjude kvaliteet on parem (teatava punktini), kui kohtvalguse valguslaik katab väiksema ala.

Filtri tüüp (Filter Type)

Kasti (Box), telgi (Tent) ja Gaussi (Gauss) tüüpi filtrid määravad ära selle, millist algoritmi kasutada puhvervarjude sakisilumiseks.

Need on otseselt seotud testkiirte (Samples) seadega: väärtuse **1** puhul on varjude filtrid välja lülitatud ning ükski neist ei avalda mitte mingisugust mõju.

Kast (Box)

Puhverdatud varjude sakisilumine toimub kastifiltri abil.

See on Blenderi originaalfilter. Kastifilter on üpris madala kvaliteediga ning seda kasutatakse madala eraldusvõimega renderdusteks, kuna see jätab väga teravad sakid. See filter kasutab ainult sama piksli piiresse langevaid puhvri andmeid, arvestamata ümbritsevate pikslite väärtusi. Kastifilter on tihti kasulik nurklike objektide ning horisontaal- või vertikaaltriipudega objektide jaoks.

Telk (Tent)

Puhverdatud varjude sakisilumine toimub telgi tüüpi filtri abil.

Telk on lihtne filter, mis annab terava tulemuse ja on üldiselt suurepärase üldotstarbeline filter. See filter võtab lõpptulemuse arvutamisel arvesse ka ümbritsevasse pikslitele vastavaid puhvri väärtusi.

Gaussi filter (Gauss)

Puhverdatud varjude sakisilumine toimub "Gaussi" filtri abil.

See filter annab väga pehme/uduse sakisilumise. Seetõttu on see filter suurepärase kõrge eraldusvõimega renderdusteks.

Erinevate sakisilumisfiltrite ning nende kasutusviiside kohta võib juurde lugeda renderdamise peatüki [sakisilumise lõigust](#).

Testpunktid (Samples)

Välja Samples väärtus võib olla **1** kuni **16**. See number määrab varjukaartide arvutamisel iga piksli kohta arvutatavate testpunktide arvu.

Kõrgema väärtuse puhul on valgusti poolt heidetud varjud filtreeritumad, siledamad ja parema sakisilumisega, ent nende arvutamiseks kulub rohkem aega ning arvuti mälu. Sakisilumiseks kasutatava meetodi määrab see, kas filtriks on valitud kast (Box), telk (Tent) või Gauss, nagu eelpool kirjeldatud.

Kui testpunktide arv väljal Samples on **1**, siis on puhverdatud varjude sakisilumine välja lülitatud.

Pehmus (Soft)

Pehmuse välja Soft väärtus võib olla **1.0** kuni **100.0**. See määrab, kui suurelt alalt puhverdatud varjude sakisilumiseks testpunkte võetakse. Mida suurem on pehmuse välja Soft väärtus, seda pehmem/sujuvam on sakisilumise tulemus heidetud varjude ääritel.

Andmepuhvrid (Sample Buffers)

Andmepuhvrite valiku Sample Buffers väärtus võib olla kas **1**, **4** või **9**, ning see määrab, kui mitut varjupuhvrit sakisilumisel kasutatakse.

Seda valikut kasutatakse erijuhtudel, näiteks väikeste liikuvate objektide puhul, mis heidavad väga pisikesi varje (näiteks kiud). Ühe piksli laiuste varjude peal töötab sakisilumine halvasti ning puhvri suurendamine valikuga Buffer Size seda ei paranda.

Selle valikuga on võimalik teha täiendav andmete keskmistamise etapp enne tavalise varjupuhvri sakisilumise algoritmi (selle, mida määravad filtrid Box/Tent/Gauss ning valikud Samples ja Soft) rakendamist.

Vaikesätte **1** puhul on see välja lülitatud.

Kõrgemad väärtused annavad ühtlasema sakisilumise, kuid vaata ette – valiku Sample Buffers väärtuse **4** korral kulutab Blender neli korda rohkem mälu ja protsessori tööaega, sest ta peab arvutama nii palju varjupuhvreid.

Korrapäratu varjugeneraator



Puhverdatud varju tüübiks on määratud korrapäratu (Irregular)

Korrapäratu (Irregular) varjude genereerimise meetod heidab teravad ja tugevad varjud, mille paigutus on sama täpne kui kiirtejälituse meetodil heidetud varjudel. See meetod on väga kiire, sest puhvrit saab genereerida paralleelselt mitmel protsessoril.

Korrapäratu varjugeneraator toetab läbipaistvaid varje. Läbipaistvate varjude puhul tuleb kõigepealt määrata varjude sätted selle objekti jaoks, millele vari langeb. (Material → Shadow → Cast Buffer Shadows (materjal -> varjud -> heida puhverdatud varje) ja Buffer Bias (puhvri nihe))

Sügav varjugeneraator

[[File:25-Manual-Lighting-Lamps-Spot-Buf-Deep.png|thumb|313px|Puhverdatud varju tüübiks on määratud sügav (Deep)]] Sügav (Deep) varjupuhver toetab läbipaistvust ning kasutab paremaid filtreid, seda kõike mälu ning töötlusaja arvelt.

Pakkimine (Compress): Sügava varjukaardi pakkimislävi.

Üldised parameetrid

Järgnevad sätted on ühised kõigile puhverdatud varjude generaatoritele.

Nihe (Bias)

Nihke (Bias) väärtus võib olla **0.001** kuni **5.0**. Nihe lisab objekti ja selle poolt heidetud varjude asukohale väikese nihke. See võib vahel olla vajalik selleks, et korrigeerida ebatäpsusi arvutustes, mis määravad, kas mingi osa objektist on varjus või mitte.

Nihke väärtuse kahandamine vähendab vahemaad objekti ja tema varju vahel. Kui nihke väärtus on liiga väike, võivad objektid tekkida artefaktid nagu jooned ja interferentsimustrid. Seda nähtust nimetatakse tavaliselt "enesevarjutamiseks" ning valiku Bias (mis ongi selleks otstarbeks) väärtuse tõstmisega see tavaliselt kaob.

Enesevarjutust võib eemaldada ka varjupuhvri suurust (Shadow Buffer Size) või varjupuhvri genereerimise meetodit vahetades: näiteks keskmistatud klassikaliseks (Classic-Halfway) või korrapäratuks (Irregular).

Enesevarjutuse häired ilmnevad pigem kumeratel kui lamedatel pindadel. Seetõttu võib paljude kumerate pindadega stseeni puhul olla tarvilik väljade Bias või Shadow Buffer Size väärtusi tõsta.

Liiga suur nihke (Bias) väärtus võib lisaks varjude objektist eemalenihutamisele põhjustada väikeste objektide varjude täielikku kadumist. Sellisel puhul võib aidata väljade nihe (Bias), varjupuhvri suurus (Shadow Buffer Size) või koonuse laius (Spot Size) väärtuste muutmise.

Nihke täpishäälestus

Nihke (Bias) väärtust saab iga [materjali](#) jaoks eraldi täpselt häälestada material konteksti Material varju paneelil Shadow asuva nihke liuguri Bias abil. Kui puhverdatud varjudega kohtvalgusti valgus langeb selle materjaliga objektile, korrutatakse varjupuhvri kasutamisel materjali nihke (Bias) väärtus valgusti nihke (Bias) väärtusega. Väärtused **0.0** ja **1.0** on üheväärsed, kumbki ei muuda valgusti algse nihke (Bias) väärtust.

Lõike algus (Clip Start) ja lõike lõpp (Clip End)

Kui stseeni lisatakse puhverdatud varjudega kohtvalgusti, ilmub 3D-vaates kohvalgusti tähisesse üks lisajoon.

Selle joone algus vastab välja Clip Start väärtusele ja lõpp välja Clip End väärtusele. Välja Clip Start väärtus võib olla **0.1** kuni **1000.0** ja välja Clip End väärtus võib olla **1.0** kuni **5000.0**. Mõlemad väärtused on Blenderi ühikutes (BU).

Lõike algus Clip Start määrab punkti, millest alates puhverdatud varjud saavad kohtvalgusti valguslaigus esineda. Kõiki varje, mis võiksid langeda ettepoole seda punkti, ignoreeritakse ja ei arvutata.

Lõike lõpp Clip End määrab punkti, millest alates kohtvalgusti valguslaigus enam puhverdatud varje ei heideta. Kõiki varje, mis võiksid langeda tahapoole seda punkti, ignoreeritakse ja ei arvutata.

Puhverdatud varjud saavad esineda ainult Clip Start ja Clip End vahelisel alal.

Clip Start ja Clip End väärtuste muutmise abil saab määrata, millises alas varje heidetakse. Clip Start ja Clip End vahemiku

muutmine võib kiirendada renderdusaega, säästa mälu ning muuta heidetavad varjud täpsemaks.

Kohtvalgusti heidetud varjude kvaliteedi hoidmiseks või tõstmiseks tasub Clip Start ja Clip End väärtused reguleerida võimalikult täpselt vastama alale, kuhu varje heita soovitakse. Minimeerides Clip Start ja Clip End väärtustega määratud vahemaad, väheneb ka ala, kus varje arvutatakse ning nende kvaliteet on seetõttu parem.

Automaatne alguslõige (Autoclip Start) ja automaatne lõpplõige (Autoclip End)

Lisaks võimalusele määrata Clip Start ja Clip End väärtuseid puhverdatud varjudega ala piiramiseks, on võimalik Blenderil lasta need väärtused automaatselt valida.

Blender teeb seda, määrates nähtavate tippude asukohad kohtvalgusti vaatepunktist lähtuvalt.

Näpunäited

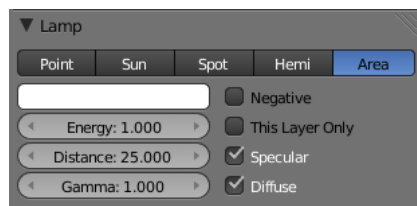
Mistahes Blenderi objekt võib 3D-vaates käituda kaamerana. Nii on võimalik valida stseenis olev kohtvalgusti ning aktiveerida kaameravaade valgusti vaatepunktist kiirklahviga Ctrl0 NumPad.

Ümber töstetud peatükki [kohtvalgustid](#)

Pindvalgus (Area)

Pindvalgus (Area) simuleerib kiirgavat valguspinda – näiteks televiisoriekraani, kaupluse neoontahvli, akent või pilvist taevast. Valguspind heidab pehmete äärtega varje, kasutades varju arvutamisel valguse allikana kasutaja poolt määratud ruudustiku punkte. See on vastandlik kunstlikele punktvalgustitele, mis heidavad teravate servadega varje.

Valikud



Ühised valikud

Ühised valikud

Vahemaa (Distance), valgusjõud (Energy) ja värvus (Color)

Need sätted on kõigil valgustitüüpidel samad ning on lähemalt kirjeldatud peatükis [Valgusti parameetrid](#).

NB! Vahemaa (Distance) väärtus mõjutab {pindvalgusi palju suuremal määral kui teisi valgustitüüpe. Enamasti saavad vahemaa (Distance) ulatuses paiknevad objektid ülesäritatud ning liigselt valgustatud. Parima tulemuse jaoks tuleks vahemaa (Distance) väärtus määrata natuke väiksemaks kui kaugus valgustatava objektini.

Gamma

Valguse heleduse gammakorrektuur. Suuremad väärtused annavad suurema kontrasti ja kiirema valguse nõrgenemiskõvera.

Pindvalgusel ei ole nõrgenemiskõvera sätteid. Ta kasutab alati kauguse ruuduga pöördvõrdelist nõrgenemiskõverat. Ainus viis pindvalguse nõrgenemiskõverat muuta on vahemaa (Distance) ja/või Gamma sätete kaudu.

Ainult sellel kihil (This Layer Only), negatiivne (Negative), läige (Specular) ja hajusus (Diffuse)

Need sätted määravad selle, kuidas valitud valgusti objekte mõjutab, lähemalt on seda selgitatud peatükis [Mida valgus mõjutab](#).

Varjud

Pindvalgusti valimisel on varjude paneel Shadow järgmiste vaikesätetega:



Adaptiivse QMC testkiirte generaatori (Adaptive QMC) sätted

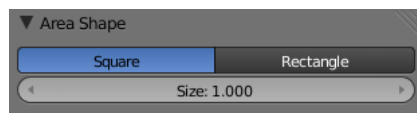


Värinaga ühtlase testkiirte generaatori (Constant Jittered) sätted

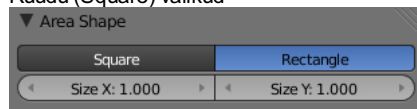
See paneel määrab kiirtejälitusega varjude sätted. Loe nende kohta lähemalt [altpoolt](#).

Pinna kuju

Pindvalguse kuju võib olla ruut (Square) või ristkülik (Rectangle).



Ruudu (Square) valikud



Ristküliku (Rectangle) valikud

Ruut (Square)/ristkülik (Rectangle)

Kiirgab valgust kas ruudu- või ristkülikukujuliselt pinnalt

Suurus (Size)/X-suurus (Size X)/Y-suurus (Size Y)

Ruudu või ristküliku mõõtmed

Kuju näpunäited

Pindvalguse sobiva kuju valimine aitab muuta loodavat stseeni usutavamaks. Näiteks võid sa soovida sisestseenis simuleerida tuppse akna kaudu langevat valgust. Selle saavutamiseks võib asetada sobivate suuruse (Size X ja Size Y) väärtustega ristkülikukujulise (Rectangle) valguspinna akna taha vertikaalselt või neoontvalguse simuleerimiseks horisontaalselt. Televiisoriekraani simuleerimiseks on tavaliselt vertikaalne ruudukujuline (Square) valguspind kõige sobilikum.

Pindvalguse kiirtejälitusega varjud



Adaptiivse QMC (Adaptive QMC) valikud

Pindvalgus saab heita varje ainult kiirtejälituse meetodil. Kiirtejälitusega heidetud varjude valikud on samad mis teistel lambitüüpidel ning on täpsemalt kirjeldatud peatükis [Kiirtejälituse parameetrid](#). Sellele lisaks on pindvalgusel mõned eripärad, mida järgnevalt kirjeldame:

Varju testkiired (*Shadow samples*)

Testkiired (Samples)

See parameeter töötab samal viisil kui kõigi teiste valgustite puhul, kuid ristkülikukujulist (Rectangle) pindvalgust kasutades on testkiirte jaoks vastavalt pindvalguse tasandi telgedele kaks eraldi seadet: X-testkiired (Samples X) ja Y-testkiired (Samples Y).

NB! Värinaga ühtlase testkiirte meetodi puhul (Constant Jittered) vastab see parameeter arvutuses kasutatavate tinglike lampide arvule. QMC algoritme kasutades käitub see parameeter samamoodi nagu punkt- ja kohtvalgustite puhul.

Testkiirte generaatorite tüübid

Adaptiivne kvaasi-Monte Carlo (Adaptive QMC); Ühtlane kvaasi-Monte Carlo (Constant QMC)

Neid üldisi sätteid kirjeldatakse peatükis [Varjude parameetrid](#).



Värinaga ühtlase meetodi (Constant Jittered) parameetrid

Värinaga ühtlane (Constant Jittered)

Pindvalgusel on ka kolmandat tüüpi testkiirte generaator Constant Jittered, mis simuleerib pigem valgustite rivi kui valguspinda. Sellel meetodil on järgmised parameetrid: täisvari (Umbra), virvtoonimine (Dither) ja värin (Jitter).

Need parameetrid on kättesaadavad ainult värinaga ühtlase generaatori (Constant Jittered) puhul ja aitavad rõhutada sellele valgustile omast pehmete varjude efekti (võimalike kvaliteedikadudega):

Täisvari (Umbra)

see parameeter suurendab täiesti varjus olevate alade varju tugevust. Valguse üleminek täisvarju ja täisvalguse vahel on kiirem (st varju ääred on teravad). Selle parameetri mõju nägemiseks peab testkiirte arv (Samples) olema vähemalt **2** või suurem.

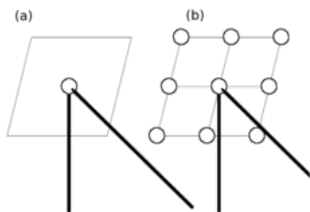
Virvtoonimine (Dither)

see parameeter laiendab testkiiri varjude servast kaugemale sarnaselt objekti servade sakisilumise nupule OSA. Virvtoonimine pehmendab kunstikult varjude ääri. Kui testkiirte arv (Samples) on seatud väga madalaks, võivad tulemused olla kehvapoolsed, seetõttu on virvtoonimist parem kasutada keskmiste välja Samples väärtustega. Kõrgete välja Samples väärtustega on virvtoonimine kasutu, kuna varjude servad on juba niigi pehmed.

Värin (Jitter)

See parameeter lisab varjudesse taustamüra, et hajutada varjude servi, nihutades testkiiri omavahel pooljuhuslikes suundades. Taas kord pole see parameeter kasulik koos kõrgete välja Samples väärtustega, kuna müra võib põhjustada pildil märgatavat teralisust.

Tehnilised detailid



Pindvalguse tööpõhimõte

Pilt "Pindvalguse tööpõhimõte" aitab mõista pehmete varjude simuleerimise protsessi.

Joonisel (a) on Blenderis defineeritud pindvalgusti. Kui valgusti on ruudukujuline, määrab varju pehmuse testkiirte Samples arv. Joonis (b) illustreerib ruudukujulist pindvalgust, mille varju ja kohtvalguse paneelil Shadow and Spot on testkiirte (Samples) arvaks seatud 3.

Sellisel juhul käsitletakse pindvalgust 3×3 ruudustikuna, kus valgusallikas on paljundatud iga sõlme peale, andes kokku üheksa valgustit.

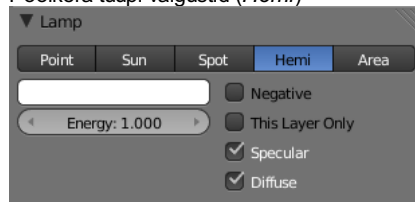
Näites (a) võrdub valgusjõud (E) $E/1$ ja näites (b) on iga üksiku valgusti valgusjõud $E/(\text{valgustite arv})$. Iga tinglik valgusallikas heidab õrna varju (võrdeliselt oma valgusjõuga) ning varjude omavaheline kattumine põhjustabki pehme varju (varjude kattumise ala on tumedam, üksiku varju ala heledam).

Näpunäited

Pindvalguse suuruse (Size) muutmine ei mõjuta valgustist kiirguva valguse tugevust. Seevastu valgusti skaala muutmine 3D-vaates kiirklahviga S võib aga valgustugevust oluliselt muuta. See on nõnda programmeeritud selleks, et ka pärast valgustuse detailset häälestamist oleks võimalik kogu stseeni mõõtkava muuta, ilma et valgustugevus selle all kannataks. Kui sa tahad muuta ainult pindvalguse suurust ilma valgustugevust muutmata, on tungivalt soovitatav kasutada valgusti suuruse (Size) parameetrit.

Värinaga ühtlase (Constant Jittered) generaatori kasutamiseks aeglasel arvutil töötades on pehmete varjude simuleerimiseks kasulik seada testkiirte (Samples) arv võimalikult madalaks (näiteks 2) ning aktiveerida kas täisvarju (Umbra), virvtoonimise (Dither) ja/või värina (Jitter) valikud. Siiski ei anna need parameetrid sama häid tulemusi kui kõrge testkiirte (Samples) arv.

Poolkera tüüpi valgustid (*Hemi*)



Poolkeravalgusti paneel

Poolkeravalgusti (Hemi) kiirgab valgust 180° poolkera kõigist suundadest, simuleerides pilvise või muidu ühtlase taeva poolt heidetavat valgust. Teisisõnu kiirgab poolkeravalgusti valgust ühtlaselt helendava, kogu stseeni ümbritseva poolkerana.

Sarnaselt päikesevalgusele (Sun) pole ka poolkeravalgusti asukoht oluline. Tähtis on ainult valgusti suund.

Poolkeravalgusti tähiseks stseenis on neli kaart, mis visualiseerivad poolkera kujulise kupli paigutust ruumis. Punktiirjoon poolkera sisemuses näitab suunda, kuhu kiirgab kõige rohkem energiat.

Seaded

Valgusjõud (Energy) ja värvus (Color)

Need sätted on kõigil valgustitüüpidel samad ning neid on lähemalt kirjeldatud peatükis [Valgusti parameetrid](#).

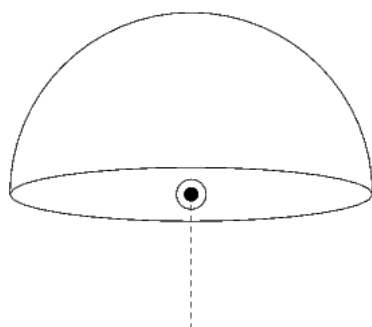
Kiht (Layer), negatiivne (Negative), läige (Specular) ja hajusus (Diffuse)

Need sätted määravad selle, kuidas valitud valgusti objekte mõjutab. Lähemalt on neid valikuid kirjeldatud peatükis [Mida valgus mõjutab](#).

Poolkeravalgustil ei ole nõrgenemiskõvera seadeid, ta kasutab alati konstantset nõrgenemiskõverat (st valgus ei nõrgene).

Kuna see on ainus valgusti, mis ei jäta varje, ei ole tema sätetes varjude paneeli Shadow.

Tehnilised detailid

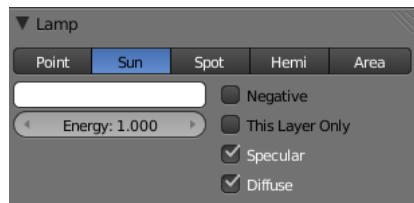


Poolkeravalgusti tööpõhimõtte joonis

Päikesevalgus (Sun)

Päike (Sun) heidab ühtlase tugevusega ühesuunalist valgust. 3D-vaates märgitakse päikesevalguse allikas musta rõngaga ümbritsetud täpiga, millest väljuvad kiired, ja punktiirjoonega, mis märgib valguse suunda. Seda suunda saab muuta, pöörates päikesevalguse allikat (Sun) nagu tavalist objekti. Kuivõrd päikesevalgus on aga ühtlane ja ühesuunaline, ei mõjuta tema asukoht stseenis renderdatud tulemust (välja arvatud juhul, kui sa kasutad "[taeva ja atmosfääri](#)" (*Sky & atmosphere*) valikuid). Päikesevalguse allika asukoht ei mõjuta mitte midagi.

Valikud



Päikesevalguse paneel

Ühised parameetrid

Valgusjõud (Energy) ja värvus (Color)

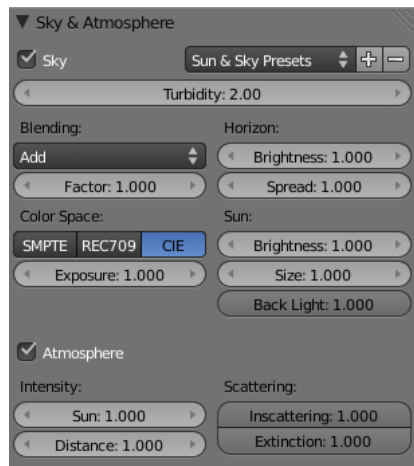
Need sätted on kõigil valgustitüüpidel samad ja neid on lähemalt kirjeldatud peatükis [Valgusti parameetrid](#).

Negatiivne (Negative), ainult sellel kihil (This Layer Only), läige (Specular) ja hajusus (Diffuse)

Need sätted määravad, milliseid objekti omadusi valgusallikas mõjutab, nagu kirjeldatud peatükis [Mida valgus mõjutab](#).

Päikesevalgusel ei ole nõrgenemiskõvera seadeid, see kasutab alati konstantset nõrgenemist (st ei nõrgene).

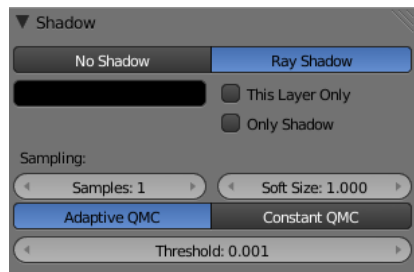
Taeva & atmosfääri paneel



Taeva & atmosfääri paneel Sky & Atmosphere

Atmosfääri ja läbi selle paistva päikese väljanägemise määramiseks on palju erinevaid sätteid. Loe lähemalt peatükist [Taevas ja atmosfäär](#).

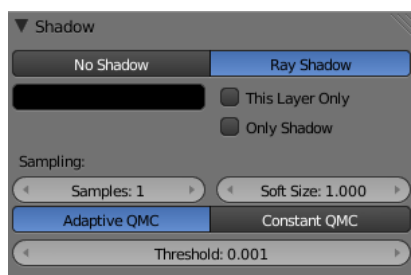
Varjude paneel



Varjude paneel Shadow

Päikesevalguse valimise järel on varjude paneel Shadow and Spot järgmiste vaikesätetega:

Päikese varjud kiirtejälitusega



Varjude paneel Shadow

Päikesevalgus saab heita varje ainult kiirtejälituse meetodil. Paljud varjudega seotud valikud on siinkohal samad kui teistel lambitüüpidel ja neid on täpsemalt kirjeldatud peatükis [Varjude omadused](#).

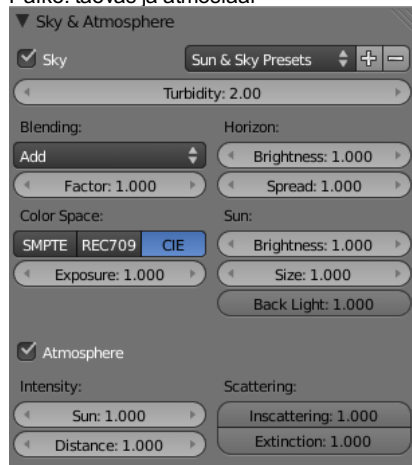
Kiirtejälituse meetodil heidetud varjudega seotud valikud on samad mis teistel lambitüüpidel ja neid on täpsemalt kirjeldatud peatükis [Kiirtejälituse parameetrid](#).

Näpunäited

Päikesevalguse allikas võib olla väga käepärane avatud ruumis ühtlase päevavalguse saavutamiseks.

Tõstetud peatükki [Doc:ET/2.5/Manual/Lighting/Lamps/Sun?](#)

Päike: taevas ja atmosfäär

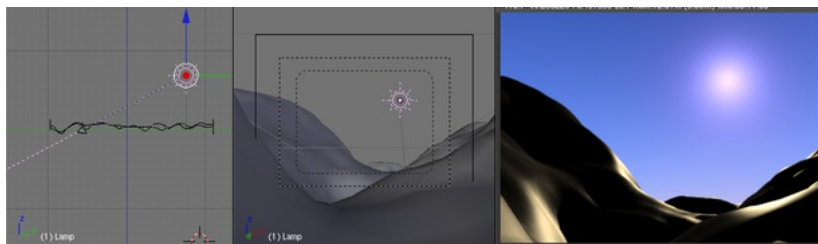


Taeva ja atmosfääri seadete paneel Sky & Atmosphere

Selle paneeli valikute abil saab sisse lülitada reaalselt taevast ja atmosfääri simuleerivad efektid: päikesekiirte hajumise kilomeetrite paksuses õhukihis. Näiteks kui päike on kõrge, paistab taevast sinine (horisondi lähedal veidi valkjas). Kui päike on horisondi lähedal, on taevast tumesinine või lillakas ja horisont muutub oranžiks. Atmosfääri hajutav toime on eriti hästi nähtav nõrga udu või vine korral: mida kaugemal mingi objekt on, seda rohkem "sulandub" tema värv helehalli tooni sisse... Mine ilusal soojal päeval loodusesse ja sa võid seda jälgida.

Selle efekti sisselülitamiseks tuleb kasutada äikesevalgust (Sun). Selle valgusti *asukoht* ei ole oluline, rolli mängib hoopis tema suund, mis määrab kellaaja (st päikese suuna taevast). Alustuseks on soovitatav taastada päikesevalguse suuna vaikeväärtus (kiirklahviga AltR või sisestades igale teisenduse omaduste paneeli Transform Properties (N) suuna (Rotation) väljale X/Y/Z väärtuse 0). See annab stseenile kena keskpäevase (troopikavööndi) päikese.

Taeva ja atmosfääri efekti juures on olulised kaks nurka: "langemisnurk" (*incidence*) - nurk valguse suuna ja X-Y tasandi vahel. See määrab "kellaaja" (nagu sa või eeldada, tähendab täisnurga all allapoole suunatud valgus "keskpäeva", täisnurga all alt üles suunatud valgus "kesköö" ja nii edasi...). Pööre ümber Z-telje määrab päikese asukoha kaamera suhtes.



Päikesevalguse allikast lähtuv punktiirjoon ("valgusjoon") lõikub kaamera fookuspunktiga.

Et omada selget pilti sellest, kus päike 3D-vaates kaamera suhtes asetseb, tuleks alati püüda saavutada seda, et valgustist lähtuv punkteeritud "valgusjoon" lõikuks kaamera vaatevälja keskpunktiga (kaamera "fookusega"), nagu näidatud pildil: Päikesevalguse allikast lähtuv punktiirjoon ("valgusjoon") lõikub kaamera fookuspunktiga. Sel viisil on kaameravaates (0 NumPad, keskmine pilt) näha, kus "virtuaalne" päike asetseb.

On oluline mõista, et valgusalika asukoht ei mängi efekti tekitamisel mingit rolli, ainus oluline parameeter on siin valguse suund. Sobiv asukoht võib lihtsalt aidata stseeni ülesehitamisel.

Valikud

Päikese ja taeva eelseadistused (Sun & Sky Presets)

- Klassikaline (Classic):
- Kõrb (Desert):
- Mäestik (Mountain):

Taevas (Sky)

Taevas (Sky)

avab taeva sätted ning kui see on sisse lülitatud, loob "taeva" koos "päikesega" ja sulandab selle taustaga vastavalt maailma (World) seadistustele.

Hägusus (Turbidity)

See on üldine säte, mis mõjutab taevast, atmosfääri ja päikest. Mida madalam on selle parameetri väärtus, seda selgem on taevast. Suuremate väärtuste korral on taevast udune. Üldiselt tekitavad madalad hägususe väärtused selge ja sinise taeva ning "väikese" päikese. Suured väärtused muudavad taeva punakaks ning tekitavad päikese ümber suure halo. NB! See on üks parameetritest, mis muudab päikesevalguse "intensiivsust". Vaata ka näiteid allpool.

Taevaspetsiifilisemad parameetrid:

Segamine (Blending)

- Esimene rippmenüü annab nimekirja erinevatest segunemisviisidest. Valitud segunemismeetod määrab selle, kuidas taevas ja päike segunevad maailma World sätetes määratud taustaga. Segunemismeetodid on samad, mida on kirjeldatud peatükis [Värvide segamissõlm](#).
- Määr (Factor)

See parameeter määrab ära, kui palju taeva ja päikese efekti maailma taustale rakendatakse.

Värviruum (Color space)

Need nupud lubavad efekti rakendamise jaoks valida erinevaid värvirume:

- CIE
- REC709
- SMPTE

Säritus (Exposure)

See lahter võimaldab muuta renderdatud taeva ja päikese säritust (väärtus **0.0** jätab särituse korrigeerimata).

Horisont (Horizon)

- Heledus (Brightness)

See parameeter määrab värvi heleduse silmapiiril. Tema väärtus võib olla vahemikus **0.0** kuni **10.0**; madalamad väärtused muudavad silmapiiri tumedaks, suuremad heledaks. Vaata ka näiteid allpool.

- Levimine (Spread)

See parameeter määrab valguse levimise horisondil. Selle väärtus võib olla **0.0** kuni **10.0**, väikesed numbrid vähendavad valguse levikut silmapiiril, suured panevad silmapiiri valguse levima üle taeva.

Päike (Sun)

- Heledus (Brightness)

Määrab päikese heleduse. Selle väärtus võib olla **0.0** kuni **10.0**, madala väärtuse juures pole päikest näha ning suurte väärtuste korral on taevas näha ainult päike.

- Suurus (Size)

See parameeter määrab päikese suuruse. Selle väärtus võib olla **0.0** kuni **10.0**, kuid pane tähele, et madalamad väärtused annavad suurema päikese ning suuremad väärtused väiksema. Päikese üldine heledus jääb siiski samaks (nagu seadistatud parameetriga heledus (Brightness), seetõttu mida suurem on päike (mida väiksem on suurus (Size)), seda rohkem päike "haihtub" taevasse (ja vastupidi).

- Taustavalgus (Back Light)

See parameeter loob päikesele hajusa taustavalguse: mida kõrgem on selle väärtus, seda rohkem on päikese ümber valgust. Selle väärtus võib olla **-1.0** kuni **1.0**. Negatiivsed väärtused ei lisa päikese ümber valgust.

Atmosfäär (*Atmosphere*)

Atmosfäär (Atmosphere)

See valik lülitab sisse atmosfääri sätteid. Atmosfäär ei muuda tausta, ent simuleerib atmosfääris esinevaid nähtusi: päikesevalguse hajumist, valguse nõrgenemist...

Valgustugevus (Intensity)

- Päike (Sun)

See parameeter määrab päikese valgustugevuse. Selle väärtus võib olla **0.0** kuni **10.0**. Kõrgemad väärtused muudavad kaugemad objektid sinisemaks.

- Vahemaa (Distance)

Selle kordaja abil teisendatakse Blenderi ühikud atmosfääriefektide jaoks sisukatesse suurustesse. Võimalikud väärtused algavad nullist, kõrged väärtused lisavad stseenile kollast valgust.

Hajumine (Scattering)

- Sisse hajumine (Inscattering)

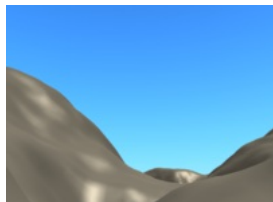
Selle parameetri abil saab vähendada kaamera ja stseeni objektide vahelisse atmosfääri sisse hajunud valguse hulka. Selle väärtus peaks olema **1.0**, kuid seda võib muuta huvitavate (ja ebarealistlike) tulemuste saamiseks.

- Neeldumine (Extinction)

Selle parameetri abil saab vähendada stseenis olevatelt objektidelt lähtuva valguse neeldumist. Selle parameetri, nagu ka sisse hajumise väärtus peaks olema **1.0**, kuid seda võib muuta, madalamad väärtused vähendavad neeldumist. Võimalikud väärtused on **0.0** kuni **1.0**.

Näited

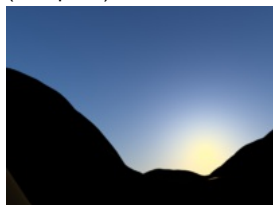
Kõigepealt vaatame, millised tagajärjed on päikese suuna muutmisel:



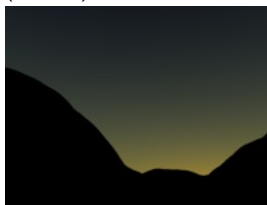
Päike otse pea kohal (keskpäev).



Päike sügaval "maa all" (kesköö).



Päike veidi horisondi kohal (päikeseloojang).

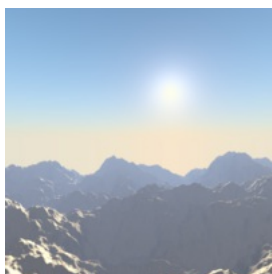


Päike veidi horisondi taga (videvik).

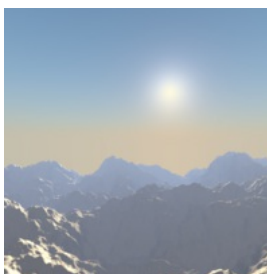
Erinevad päikese suunad. Suurus (Sun Size) **5.0**, ülejäänud vaikesätted.

[Näidetes kasutatud 2.4 versiooni .blend-faili](#).

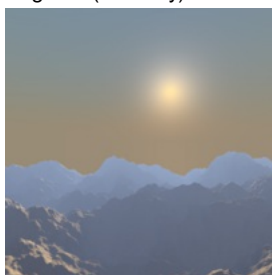
Erinevate parameetrite mõju (näidetes kasutatud [seda 2.4 .blend-faili](#)):



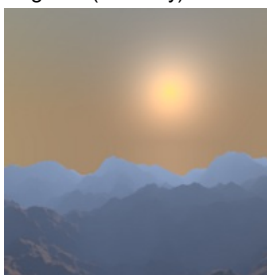
Hägusus (Turbidity): **2.0**.



Hägusus (Turbidity): **2.3**.



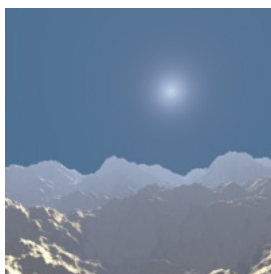
Hägusus (Turbidity): **5.0**.



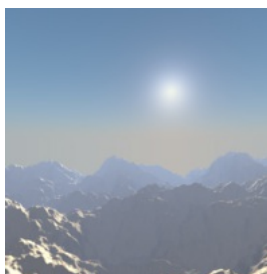
Hägusus (Turbidity): **10.0**.

Hägususe (Turbidity) erinevad väärtused, ülejäänud vaikesätted.

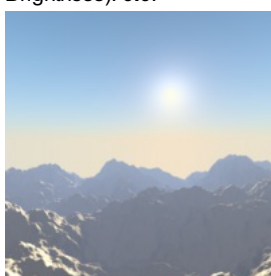
Taevas (Sky)



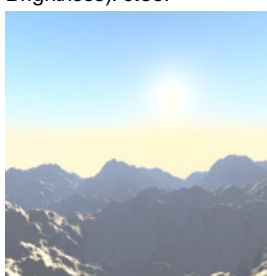
Silmapiiri heledus (Horizon Brightness): **0.0**.



Silmapiiri heledus (Horizon Brightness): **0.85**.



Silmapiiri heledus (Horizon Brightness): **1.0**.

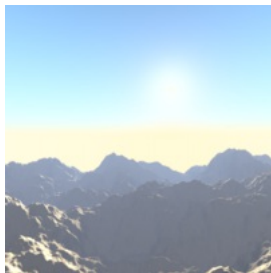


Silmapiiri heledus (Horizon Brightness): **1.85**.

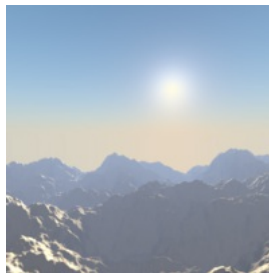
Brightness): **1.04.**

Brightness): **1.13.**

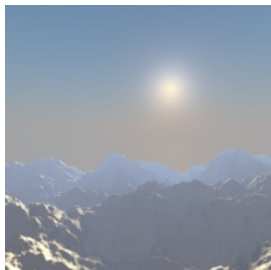
Silmapiiri heleduse (Horizon Brightness) erinevad väärtused, ülejäänud vaikesätted.



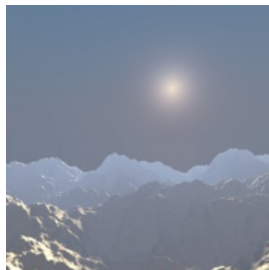
Silmapiiri levimine (Horizon Spread): **0.7.**



Silmapiiri levimine (Horizon Spread): **1.2.**

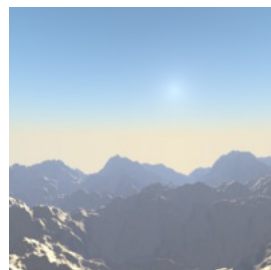


Silmapiiri levimine (Horizon Spread): **2.2.**

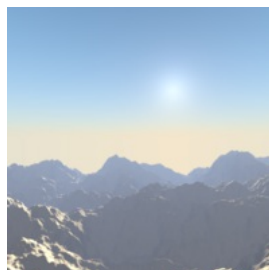


Silmapiiri levimine (Horizon Spread): **5.0.**

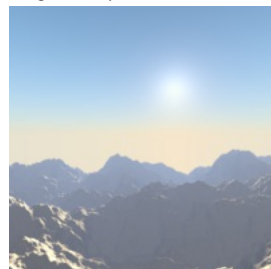
Erinevad silmapiiri levimise (Horizon Spread) väärtused, ülejäänud vaikesätted.



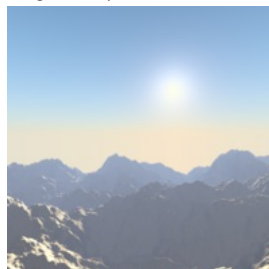
Päikese heledus (Sun Brightness): **0.2.**



Päikese heledus (Sun Brightness): **0.5.**

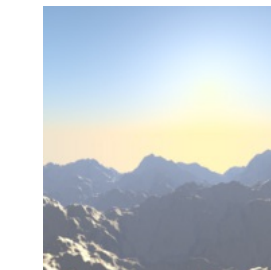


Päikese heledus (Sun Brightness): **0.75.**

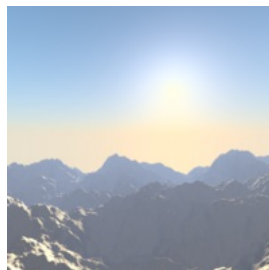


Päikese heledus (Sun Brightness): **1.0.**

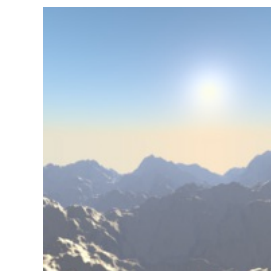
Erinevad päikese heleduse (Sun Brightness) väärtused, ülejäänud vaikesätted.



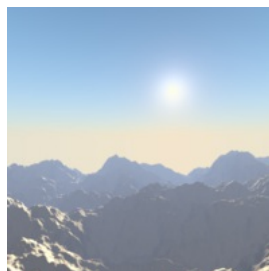
Päikese suurus (Sun Size): **2.0.**



Päikese suurus (Sun Size): **4.0.**

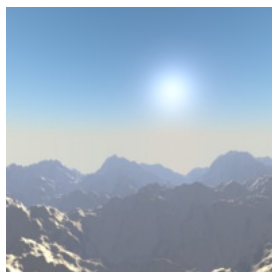


Päikese suurus (Sun Size): **7.0.**

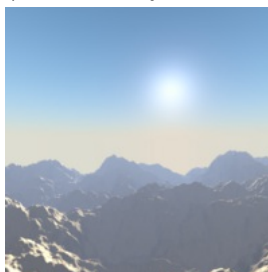


Päikese suurus (Sun Size): **10.0.**

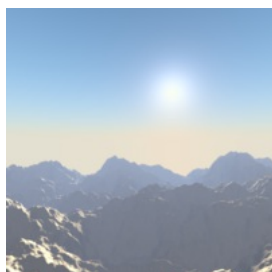
Erinevad päikese suuruse (Sun Size) väärtused, ülejäänud vaikesätted.



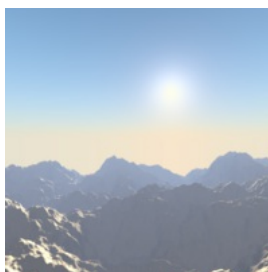
Taustavalgus (Back Light): -
1.0.



Taustavalgus (Back Light): -
0.33.



Taustavalgus (Back Light):
0.33.

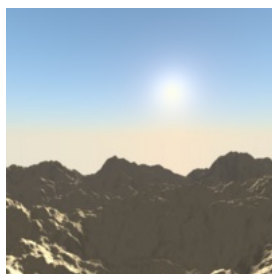


Taustavalgus (Back Light):
1.0.

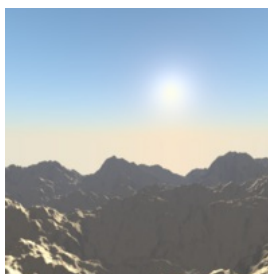
Erinevad taustavalguse (Back Light) väärtused, päikese heleduse (Sun Bright) väärtus **2.5**, ülejäänud vaikesätted.

Atmosfäär (*Atmosphere*)

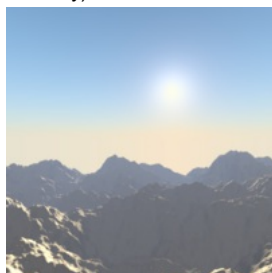
Kõigil järgnevatel renderdustel on horisontaalse heleduse (Hor.Bright) väärtus **0.2** ja päikese heleduse (Sun Bright) väärtus **2.0**.



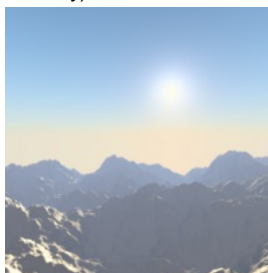
Päikese valgustugevus (Sun
Intensity): **1.0.**



Päikese valgustugevus (Sun
Intensity): **3.33.**

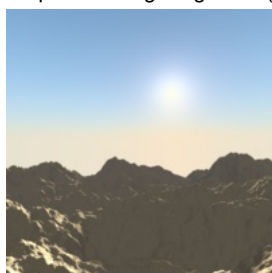


Päikese valgustugevus (Sun
Intensity): **6.66.**

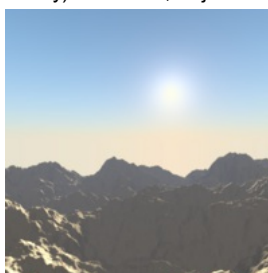


Päikese valgustugevus (Sun
Intensity): **10.0.**

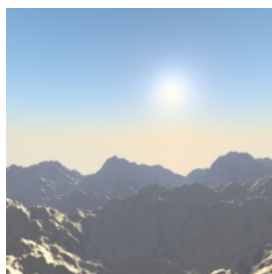
Erinevad päikese valgustugevuse (Sun Intensity) väärtused, ülejäänud vaikesätted.



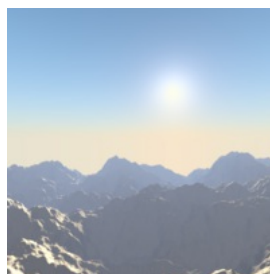
Sisse hajumine
(Inscattering): **0.1.**



Sisse hajumine
(Inscattering): **0.33.**

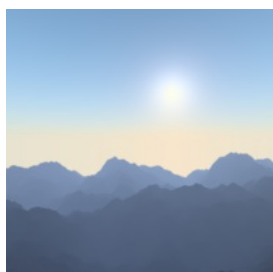


Sisse hajumine
(Inscattering): **0.66**.

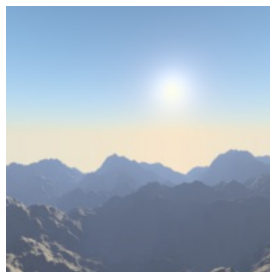


Sisse hajumine
(Inscattering): **1.0**.

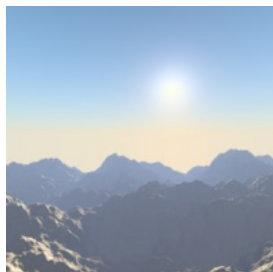
Erinevad sissehajumise Inscattering väärtused, ülejäänud vaikesätted.



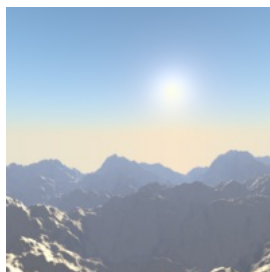
Neeldumine (Extinction): **0.0**.



Neeldumine (Extinction):
0.33.

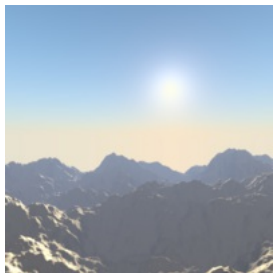


Neeldumine (Extinction):
0.66.

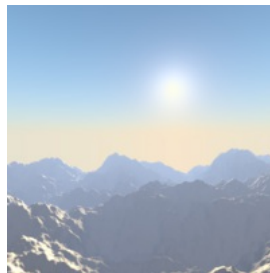


Neeldumine (Extinction): **1.0**.

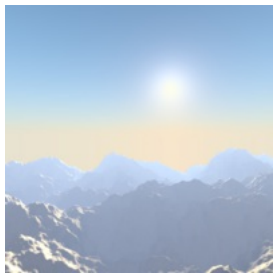
Erinevad neeldumise Extinction väärtused, ülejäänud vaikesätted.



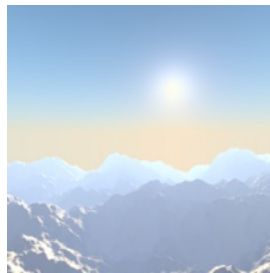
Vahemaa (Distance): **1.0**.



Vahemaa (Distance): **2.0**.



Vahemaa (Distance): **3.0**.



Vahemaa (Distance): **4.0**.

Erinevad vahemaa Distance väärtused, ülejäänud vaikesätted.

Näpunäited ja piirangud

Et päike oleks alati suunatud kaamera keskmesse, võid sa lisada päikese objektile [trajektooriiraaja \(TrackTo\)](#), seades sihiks kaamera ning suunatud teljeks ("To")-Z-telje. (Vertikaalseks teljeks ("Up") sobivad nii X kui Y). Nii saab päikese kõrguse ja suuna muutmiseks renderdatud pildil lihtsalt muuta päikese asukohta, valguse suund muutub automaatselt trajektooriiraaja tõttu. Liikuva kaamera puhul tuleks päikesele lisada ka [asukoha kopeerija \(Copy Location Constraint\)](#), seades sihiks kaamera ning lülitades sisse valiku nihe (Offset). Nii ei muutu päikese valguse suund kaamera liikumise tõttu.

Kasutades segamismeetodina vaikesättena antud liitmist (Add), tuleks maailma värviks määrata väga tume sinine, et ööstseenid näeksid õiged välja.

See efekt töötab hästi poolkeravalgusti (Hemi) või kaudvarjudega, mis lisavad veidi heledust päikese varjudesse.

Atmosfääri algoritm annab praegu peegelduste ja murdumiste puhul valesid tulemusi ning töötab ainult läbipaistmatutel pindadel. Selle arendamine jätkub järgmistes Blenderi versioonides.

Valgusvihud (*Volumetric Lightning*)

Tsiteerides Wikipediat on [valgusvihkude](#) seletus:

«Ruumiline valgus on 3D-arvutigraafikas kasutatav meetod renderdatud stseenile valgusefektide lisamiseks. See muudab ruumi läbivad valguskiired vaatajale nähtavaks. Näide ruumilisest valgusest on läbi avatud akna ruumi tungivad päikesekiired, 3D-renderduses tuntud ka kui "God Rays" (jumala kiired). Mõiste ise näib pärinevat kinematograafiast, kuid on nüüd laialdaselt kasutusel 3D-modelleerimises ja renderdamises, eriti 3D-arvutimängudes. Ruumilise valguse kasutamisel modelleeritakse valgusallikast lähtuv valguskoonus läbipaistva ruumilise objektina (mis on siis selle valguse "mahuti"). Selle tulemusena on võimalik kujutada, kuidas valgusallikast lähtuvad kiired läbivad seda mahutit täitvat kolmemõõtmelist ainekogumit (näiteks udu, tolm, suits või aur) just nagu reaalses maailmas. »

Klassikaline näide on liikuv prožektor mille valgus paistab küljelt helendava liikuva valguskoonusena.

Vaikimisi Blender valgusallikatele valgusvihke ei modelleeri. Näiteks kui Blenderis valgustada midagi kohtvalgustiga (Spot), näed sa valgustatud objekte ja pörandat nende ümber, kuid mitte helendavat vihku, mis lähtub valgusallikast ja suundub teel hajudes objektide poole.

Reaalses maailmas tekitab helendava vihu valguse hajumine õhus leiduvatel osakestel: osa sellest hajunud valgusest jõuab vaataja silma ning tekitab mulje helendavast vihust või triibust. Valguse hajumist on Blenderis võimalik simuleerida mitmel viisil, kuid ükski neist pole vaikimisi sisse lülitatud.

[Kohtvalgusti \(Spot Light\)](#) on ainus Blenderi valgusallikas, mis saab tekitada valgusvihke (kuigi osasid päikesevalguse (Sun) "[taeva ja atmosfääri](#)" [efekte](#) võib ka lugeda valgusvihkudeks).

Näide

[Blenderi animatsioon kohtvalgustist.](#)
[\[video link\]](#)

Vaata ka

- [Udu](#)
- [Suits](#)
- [Ruumilised materjalid](#)

Kaudvari (*Ambient Occlusion*)

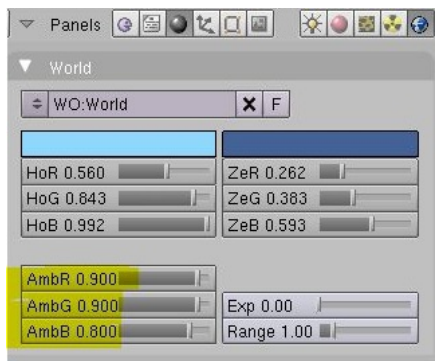
Kaudvari (AO) on keeruline kiirtejälitusel põhinev arvutus, mis simuleerib pehmeid üldvalguse varje, võltsides (luues samase efekti ilma täpset füüsikalist mehhanismi järgimata) tajutava pimeduse sisenurkades, võre lõikepunktides, vagudes ja pragudes ning mujal, kus üldvalguse juurdepääs on piiratud või blokeeritud.

Reaalses maailmas sellist asja nagu kaudvari ei eksisteeri. See on füüsikaliselt ebatäpne (kuid üldiselt hea väljanägemisega) renderdustrikk. Selle tööpõhimõte on luua iga objekti külge ümbritseva tipu ümber poolkera, määrata kindlaks, milline osa sellest poolkerast on teiste objektide poolt varjatud, ja varjutada piksleid vastavalt.

Valgusega pole kaudvarjul mitte mingit pistmist. See on puhtalt renderdustrikk, mis näeb hea välja, kuna tavaliselt on ka reaalses maailmas üksteisele lähedal asuvad pinnad (nagu väikesed praod) varjude ning neisse koguneva mustuse tõttu siledatest pindadest tumedamad.

Kaudvari ei simuleeri valguse pörkamist või objektidest läbikumamist, vaid hindab võimalikku tulemust ligilähedaselt. Selle tõttu töötab kaudvari ka siis, kui stseenis ei ole ühtegi valgustit. Ainult kaudvarju kasutamine on üpris vilets viis stseeni "valgustamiseks".

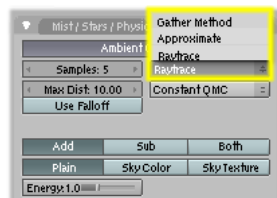
Kaudvarju kasutamiseks peab renderduskonteksti Render varjutuspaneelis Shading olema sisse lülitatud kiirtejälitus (*Ray Tracing*).



Maaailma (World) paneel esiletõstetud üldvalguse värviliuguritega (Blender 2.4).

Üldvalguse (*Ambient*) värviliugurid tuleb seadistada vastavalt oma soovile. Vaikesättena on maailma üldvalguse värviks must, simuleerides keskoist keldrit elektrikatkestuse ajal. Musta kasutamine üldvalguse värvina tumestab kõiki värve. Hea keskpäevane vabaõhuvärv on RGB (**0.9, 0.9, 0.8**) – päikeseline helekollane värv helgel, ent mitte kiiskaval päeval.

Valikud



Kaudvarju (Amb Occ) paneel, meetodi valik (Blender 2.4).

Faktor (Factor)

Kaudvarju tugevus – liitmis- või lahutamistehte korrutaja.

Kaudvari komposiitatakse renderdamise käigus. Kasutusel on kolm segunemisviisi:

Liitmine (Add)

Piksel saab valgust sõltuvalt tõkestamata kiirte arvust. Stseen on heledam. See simuleerib üldvalgustust.

Korrutamine (Multiply)

Pinna heledus korrutatakse kaudvarjuga, muutes objektid tumedamaks.

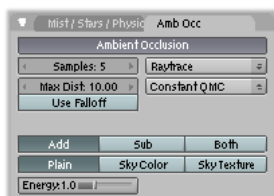
Note

Valides segunemisviisiks korrutamise (Multiply), peab stseenis olema ka valgusallikaid, muidu paistab kõik kottpime. Ülejäänud segunemisviiside puhul paistab stseen valgustatud isegi valgustite puudumisel, piisab kaudvarju efektist. Kuigi vahel peetakse kaudvarju kiireks stseeni valgustamise viisiks, on ainult selle kasutamise tulemus lame ja sumbe nagu pilvisel päeval. Enamikel juhtudel on parim viis valgustada stseen Blenderis leiduvate standardvalgustitega ning sellele lisaks aktiveerida kaudvari korrutamise segunemisviisiga ("Multiply"), et lisada detaile ja väikevarje.

Sätted kaudvarju kvaliteedi reguleerimiseks on kogumispaneelis (Gather). Need sätted kehtivad ka keskkonnavalgustusele ja kaudsele valgustusele.

Kaudvarjul on kaks peamist arvutusmeetodit: kiirtejälitus (Raytrace) ning hindamine (Approximate).

Kiirtejälitus

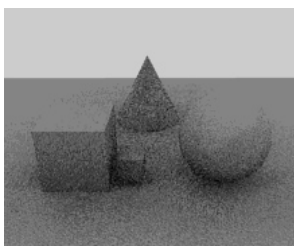


Kaudvarju (Amb Occ) paneel,
Kiirtejälituse meetod
(Raytrace) (Blender 2.4).

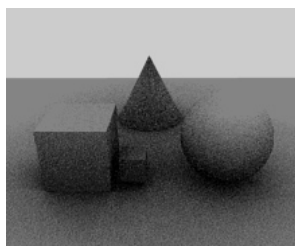
Kiirtejälituse meetod (Raytrace) annab täpsema, ent ka mürarikkama (teralisema) tulemuse. Pikema renderdusaja arvelt on siiski võimalik saavutada peaaegu müravaba pilt. See on ainus võimalus taeva tekstuuri värvide kasutamiseks.

Proovide arv (Samples)

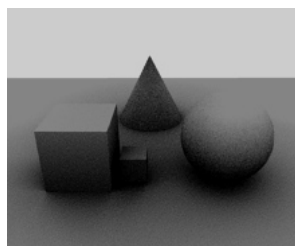
Objekti varjutuse määramiseks kasutatav kiirte arv. Suurem proovide arv annab pikema renderdusaja arvelt sujuvama ja täpsema tulemuse. Vaikesäte **5** on tavaliselt piisav eelvaadeteks. Tegelik kasutatavate kiirte arv on selle numbri ruut (Samples väärtus **5** tähendab **25** kasutatud kiirt). Kiired heidetakse poolkera suunas pseudojuhusliku mustrina (mille määravad allpool kirjeldatud proovide võtmise meetodid). Selle tulemusena võib kõrvuti asetsevatel pikslitel olla erinev varjutusaste, kui just katses kasutatavate kiirte arv ei ole piisavalt suur usaldusväärse statistilise keskmise leidmiseks.



Kaudvari proovide arvu
(Samples) väärtusega **3**.

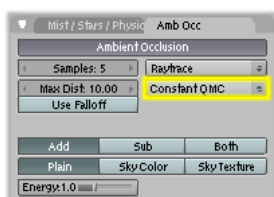


Kaudvari proovide arvu
(Samples) väärtusega **6**.



Kaudvari proovide arvu
(Samples) väärtusega **12**.

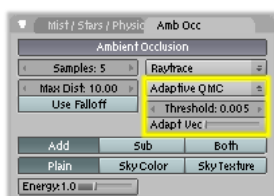
Varjutusproovide võtmiseks on kolm standardvalikut:



Ühtlase kvaasi-Monte Carlo
meetodiga (Constant QMC)
proovid.

Ühtlane kvaasi-Monte Carlo (Constant QMC)

Tavaline kvaasi-Monte Carlo meetod, mis annab ühtlaselt ja juhuslikult jaotatud kiired.



Adaptiivne varjutus Monte
Carlo algoritmi meetodil
Adaptive QMC.

Adaptiivne kvaasi-Monte Carlo (Adaptive QMC)

Täiendatud QMC versioon, mis hindab, kas osasid katseandmeid võib arvutusest välja jätta, sõltuvalt järgmistest sätetest:

Lävi (Threshold)

See määrab piiri, millest allpool katsepunkt määratakse täiesti tõkestatuks ("must") või tõkestamatuks ("valge"), ning rohkem proove ei arvutata.

Kiirusega kohandamine (Adapt to Speed)

See vähendab kaudvarju proovide hulka kiiresti liikuvate pikslite puhul. Kuna see kasutab kiirusvektori (Vector) renderduskäiku (*Render Pass*), peab see olema sisse lülitatud (loe lähemalt [renderduskäikude peatükist](#)).

QMC algoritmi meetodist

Loe Quasi-Monte Carlo meetodist lisa ka [kiirtejälituse meetodil heidetud varjude peatükist](#).

Värinaga ühtlane (Constant Jittered)

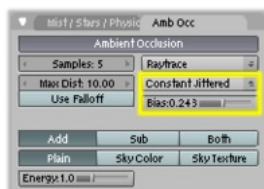
See on ajalooline katseandmete kogumise meetod, mis kipub tekitama kindlasuunalisi artefakte.

Kitsendamine (Bias)

Proovikiirte kogumise kupli (poolkera) alusnurk (radiaanides) muudetakse kitsamaks (kuppel ei ole enam poolkera – tema ristlõige ei ole enam poolring, vaid ringjoone kaar nurgaga “ π - Bias”). Kitsenduse abil saab kontrollida, kui siledad on “silutud” normaalidega võred kaudvarjuga renderduses. Kuna kaudvarju arvutamiseks kasutatakse tegelikku, tahulist võret, võivad kaudvarjus isegi “silutud” tipunormaalidega objektidel külgedevahelised nurgad paistma hakata. See juhtub tänu kaudvarju proovikiirte võtmise meetodile, mida saab kitsendamise liuguriga Bias kontrollida. Pane tähele, et kuigi see võib juhtuda ka QMC meetodi puhul, on ta oluliselt silmatorkavam värinaga ühtlase (Constant Jittered) proovide võtmise meetodi puhul. Ja igal juhul ei ole QMC meetodi puhul ka võimalik kitsenduse (Bias) valikut kasutada...

Nõrgenemine (Attenuation)

Tõkestuse arvutamise kiirte pikkus. Mida pikem on see vahemaa, seda suurem mõju on kaugel asuval geomeetria kaudvarjule. Suur kauguse (Distance) väärtus tähendab ka seda, et renderdusmootor peab tõkestavaid objekte otsima suuremalt alalt – seetõttu hoiab madalam väärtus kokku renderdusaega ning tuleks optimeerida nii väikeseks, kui soovitud efekti saavutamiseks vähegi võimalik.



Värinaga ühtlane (Constant Jittered) meetod.

24×24 UV kera vaikesättega Liuguri Bias tõstmine Bias: **0.05**. Pane tähele kera väärtuseni **0.15** eemaldab pinnal nähtavaid servi, kuigi servade defekti. sätete põhjal peaks ta olema sile.

Hindamine (Approximate)



Kaudvarju (Amb Occ) paneel, ligilähedase hindamise meetod Approximate.

Ligilähedase hindamise meetod (Approximate) annab sama renderdusaja puhul palju siledama tulemuse, kuid nagu ka nimi ütleb, on tegu ainult kiirtejälituse meetodi ligilähedase jäljendamisega, mis võib tekitada pildidefekte ning pole võimeline kasutama taeva tekstuuride põhivärvi.

Ligilähedase hindamise meetod kipub tihti peale objekte üleliia varjutama. Selle vältimiseks on meetodil kaks teineteist täiendavat lisavalikut:

Käigud (Passes)

Eeltöötluse käikude arv **0** (ilma eeltöötluseta) kuni **10**. Ma ei tea, mida see eeltöötlus täpselt teeb...

Viga (Error)

See on ligilähedase hindamise meetodi veasallivuse tegur (teisisõnu maksimaalne lubatud erinevus ligilähedaselt hinnatud ja täpselt arvutatud tulemuse vahel). Mida madalam on selle väärtus, seda aeglasem on renderdusprotsess, ent seda täpsemad tulemused. Väärtus võib olla **0.0** kuni **10.0**, vaikesättena **0.250**.

Pikslite puhvermälu (Pixel Cache)

Puhvermälu jätab juba arvutatud pikslite väärtused meelde ning kasutab seda naaberpikslite interpoleerimiseks. See valik muudab renderdusprotsessi kiiremaks ning üldiselt ei vähenda nähtavalt pildi kvaliteeti.

Parandus (Correction)

Korrigeeriv tegur liigse varjutuse vähendamiseks. Selle väärtus võib olla **0.0** (ilma korrigeerimisega) kuni **1.0**.

Ühised parameetrid

Nõrgenemiskõver (Falloff)

Kui see on aktiveeritud, siis tõkestavate objektide kaugus mõjutab varjude “sügavust”. See tähendab, et mida kaugemal on tõkestav objekt, seda heledam on selle “vari”. See efekt ilmneb ainult siis, kui tugevuse lahtri Strength väärtus on suurem kui **0.0**. See efekt matkib valguse hajumist atmosfääris...

Tugevus (Strength)

See väärtus määrab varjude nõrgenemise kiiruse, kui Use Falloff on sisse lülitatud. Mida kõrgem on selle väärtus, seda lähemad on varjud tänu kiiremale nõrgenemisele (näiteks tolmusemas keskkonnas). Väärtus võib olla **0.0** (vaikesäte, nõrgenemist ei toimu) kuni **10.0**.

Tehnilised detailid

Kaudvarju arvutatakse pinna punktidest väljuvate kiirte järgi, lugedes, kui paljud neist jõuavad taevasse ja kui paljusid tõkestab mõni teine objekt.

Valguse määr teatud punktis on võrdeline sellega, kui palju sellest punktist lähtuvaid kiiri jõuab taevani. Igast pinna punktist vallandatakse poolkerana kõikides suundades teatud arv kiiri. Kui kiir tabab mõnd teist pinda (on tõkestatud), loetakse seda kiirt varjuks, vastasel juhul loetakse kiirt valguseks. "Valguse-" ja "varjukiirte" suhtarv määrabki lähtepunktis asuva piksli heleduse.

Näpunäited

Kaudvari on kiirtejälituse meetodil põhinev tehnika (vähemalt kasutades kiirtejälituse (Raytrace) meetodit), seetõttu on ta aeganõudev protsess. Lisaks sellele sõltub jõudlus ka ruumi jaotamise puu (*octree*) suuruselt, loe lähemalt [renderdamise peatükist](#).

Keskkonna valgus (*Environment Lighting*)

Keskkonnavalgus annab valgust igast suunast. Valgus arvutatakse kiirtejälituse meetodil, mida kasutatakse ka kaudvarju (*Ambient Occlusion*) puhul. Erinevus seisneb selles, et keskkonnavalgus võtab arvesse materjali varjutuse sätete all olevat keskkonna (*ambient*) parameetrit, mis määrab selle, palju üldvalgust/värvust materjal vastu võtab. Samuti saad sa valida keskkonna värvuse allika (valge, taeva värv, taeva tekstuur) ja valgusjõu.

Proovi mõlemat sätet samaaegselt kasutada, et paremat globaalset valgustust saavutada.

See on sobiv välistseenide valgustuses taeva simuleerimiseks. Keskkonnavalgus võib tihti peale olla üpris mürarikas.

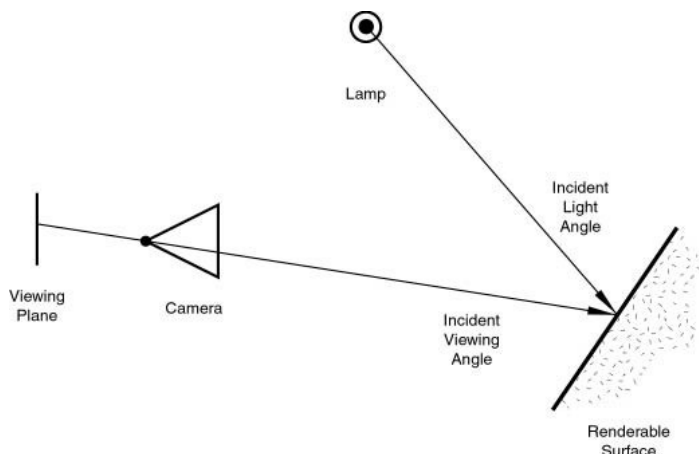
Sissejuhatus materjalidesse

Enne kui õpid materjale tõhusalt kasutama, pead mõistma, kuidas simuleeritavad valgused ja pinnad omavahel Blenderi renderdusmootoris läbi saavad ning kuidas materjali sätteid neid suhteid määravad. Selle mootori süviti tundmine aitab sul sellest viimast võtta.

Blenderiga loodud renderdatud pilt on stseeni projektsioon kujuteldavale pinnale nimega *vaatetasand*. Vaatetasand on sarnane tavalise kaamera filmilindile või inimese silma kepikestele ja kolvikestele, kuid reaalse valguse asemel võtab see vastu simuleeritud valgust.

Stseenist pildi renderdamiseks peame me kõigepealt kindlaks tegema, milline valgus igale vaatetasandi punktile jõuab. Parima vastuse saamiseks tuleks tõmmata sirge joon (simuleeritud valguskiir) tagasi läbi selle vaatepinna punkti ja läbi fookuspunkti (kaamera asukoht), kuni see jõuab mingi stseenis oleva renderdatava pinnani; seejärel saame määrata, milline valgus tolele punktile langeks.

Selle pinna omadused ning valguse langusnurk ütlevad meile, kui palju valgusest mööda vaatenurka edasi peegeldatakse (vt pilti *Renderdusmootori põhiprintsiibid*).



Renderdusmootori põhiprintsiibid.

Igal pinna punktil toimivad kaks lihtsat fenomeni, kui sellele langeb valguskiir: hajumine (*diffusion*) ja läikepeegeldus (*specular reflection*). Hajusus ja läikepeegeldus erinevad üksteisest peamiselt selle poolest, milline on nende suhe valguse langusnurga ja peegeldatava valguse nurga vahel.

Objekti varjutamine (ehk värv) võtab renderdamise ajal arvesse baasvärvi (mida mõjutavad hajusus ja läikepeegeldus) ning valguse intensiivsust.

Blenderi sisseehitatud kiirtejälituse kasutamisel võivad toimida ka muud (ja keerulisemad) nähtused. Kiirtejälitusega peegelduste puhul annab pinna see punkt, millele valguskiir langes, edasi seda ümbritseva keskkonna värvi, vastavalt materjali peegelduse väärtusele (mis segab kokku baasvärvi ja ümbritseva keskkonna värvi) ning vaatenurgale.

Teisalt annab see punkt, millele valguskiir langes, kiirtejälitusega valguse murdumise puhul edasi enda taha jääva keskkonna värvi vastavalt materjali läbipaistvuse väärtusele (segades kokku baasvärvi ja taustakeskkonna oma võimalike filtreerimisväärtustega) ja võimaliku materjali murdumise väärtuse, mis muudab vaatenurka.

Renderdamise ajal sõltub valguskiire poolt valgustatud objekti varjutamine loomulikult kõigi nende nähtuste koosmõjust. Renderdatud objekti välimus sõltub paljudest omavahel seotud seadistustest:

- Maailm (keskkonna värv (*Ambient color*), kiirgavus (*Radiosity*), kaudvari (*Ambient Occlusion*))
- Valgustid
- Materjali sätteid (kaasa arvatud keskkond, kiirgavus ning kõik teised tolle kontekstpaneeli seaded)
- Tekstuur(id) ja see, kuidas nad on omavahel segatud
- Materjalisõlmed
- Kaamera
- Vaatenurk
- Vaatele ette jäävad ja poolläbipaistvad takistused
- Varjud teistelt läbipaistmatult/läbipaistvatelt objektidelt
- Renderdussätteid
- Objekti mõõtmised (pinnajagamise sõltub mõõtmetest)
- Objekti kuju (murdumised, Fresneli efektid)

Varjutajad (*Shaders*)

Materjalidel võib olla väga erinevaid omadusi. Nende kombinatsioon määrab selle, milline on materjali välimus ning kuidas varjutaja renderdamisel käitub.

Hajusvarjutus (*Diffuse Shading*)

Esimese sammuna lisa hajusvarjutuse info. Hajusvarjutust näed mattide või poolmattide objektide puhul või pindadel, millel pole heleduse laiike. See kujutab pinna mikroskoopilisi erisusi, mis paiskavad sisse tuleva valguse väga erinevate nurkade all laiali.

Läikevarjutus (*Specular Shading*)

Arvutigraafikas saad sa objekte läike (*specularity*) abil läikima panna. Läike kujutab eredate objektide peegeldusi, mis mõnda pinda tabavad. On oluline märkida, et tavapäraselt pole läikelaigud valguste tegelikud peegeldused. Kuna valgustid on tavaliselt palju

eredamad kui ümbritsev keskkond, saame neid peegeldusi aga läikelaikudega võltsida.

Peegelduvus (Reflectivity)

Et näha tõeliseid ümbritsevate objektide peegeldusi, saavad varjutajad olla kiirtejälituse (*raytracing*) nimelise tehnika abil peegeldavad. Kiirtejälitus heidab kaamerast virtuaalseid kiiri ning jälitab siis seda, kuhu need peegelduvatelt objektidelt põrgates välja jõuavad. Mitte kõik peegeldused ei ole täiuslikud nagu peegel või kroomitud kera. Varjutajad saavad simuleerida ebatäiuslikke peegeldusi, mis muudavad peegeldatud pildi uduseks. See on renderdusmootoritele aga raske, sest vajab suurt hulka arvutusi.

Läbipaistvus (Transparency)

Paljud pinnad on läbipaistvad või pooleldi läbipaistvad. Varjutaja võib luua läbipaistvust, vähendades lihtsalt oma katvust, mille tulemusena paistavad tagumised objektid läbi, kuid läbipaistavad objektid töötavad harva sedamoodi. Ka läbipaistvus saab kasutada kiirtejälitust. Enamus läbipaistvaid objekte tekitavad murdumist ehk refraktsiooni (*refraction*), mis tähistab valguskiirte painutamist, kui need erinevate refraktsiooni indeksiga (*index of refraction*, IOR) objekte läbivad. Täida klaas veega ja vaata, kuidas refraktsioon toimib. Läbipaistvust saab samuti uduseks muuta, nagu jääklaasi või mõnede plastpakendite puhul. Udused refraktsioonid nagu udused peegeldusedki on protsessorinõudlikud.

Läbikumavus (Translucency)

Kui materjal on võimeline valgust läbi laskma, aga see paiskub sedavõrd palju laiali, et udune läbipaistvus ei ole enam praktiline, võivad varjutajad simuleerida läbikumavust. Läbikumavus on hea õhukeste objektide, näiteks lehtede, paberi või lillede puhul. Arvutigraafikas arvutatakse läbikumavuse jaoks pinna tagumise külje varjutus ning see, kuidas valgus teisele küljele edasi liigub. Paksemate objektide puhul on soovitatav kasutada pinna sisehajumist.

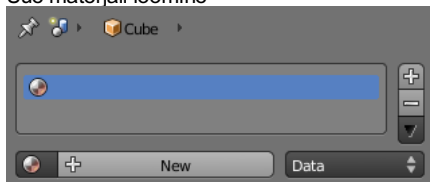
Pinna sisehajumine (Subsurface Scattering)

Pinna sisehajumine (*Subsurface Scattering* ehk sss) on efekt, mis saab simuleerida orgaanilisi materjale, mille puhul valgus suudab pinnast läbi tungida, kuid hakkab pinna all pörkuma ja paiskub laiali ning väljub teises kohas. See on tüüpiline selliste asjade puhul nagu inimnahk, küünlad, juust, viinamarjad, marmor jms. Pinna sisehajumine saab muuta materjalid tükki maad realistlikumaks, kuid see võib nõuda väga suurt renderdaja jõudlust.


Hõõgumine (Incandescence)

Mõned objektid on või näivad valgusallikatena (nt lambipirn või teleekraan). Selle saavutamiseks tuleb varjutaja info hulka vähendada.

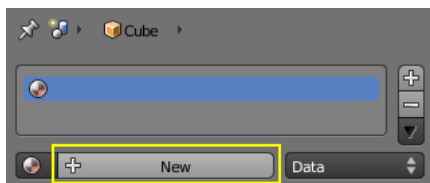
Uue materjali loomine



Materjali menüü.

Uue objektiga ei seota ühtki materjali. Vajutades nupule , sisened sa varjutamise konteksti ning ilmub materjali nuppude aken. Antud hetkel peaks see aken olema peaaegu tühi.

Uut materjali saab lisada, kasutades pildil *Uue materjali lisamine* näidatud nuppu. Valida saab järgmiste variantide vahel:



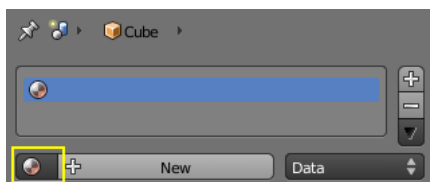
Uue materjali lisamine.

Add New

Lisa uus materjal ning lingi see aktiivse objekti või objekti andmetega. Nagu teistegi andmeblokkide puhul, nimetab Blender uue materjali automaatselt Material.001-ks ning jätkab nummerdamist sealt edasi.

Materjalidele selgete nimede andmine on väga hea mõte, eriti kui neid lingitakse mitmete objektidega.

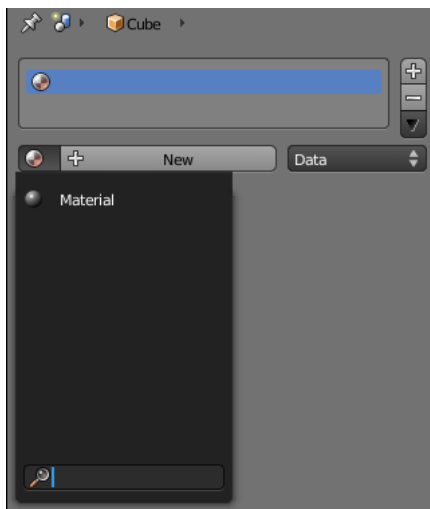
Materjali jagamine mõne teise objektiga



Olemasoleva materjali valimine.

Vali olemasolev materjal

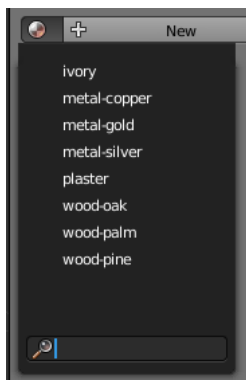
Vali nimekirjast juba olemasolev materjal.



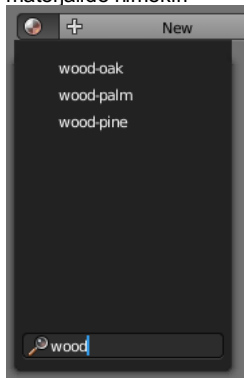
Olemasoleva materjali rippmenüü

Blender on ehitatud nii, et saaksid *kõike* taaskasutada, sealhulgas ka materjalide sätteid erinevate objektide vahel. Duplikaat materjalide loomise asemel saad lihtsalt uuesti kasutada juba olemasolevat materjali. Selleks on kaks võimalust:

- Kui võre on valitud, vajuta materjali nimest vasakule jäävale kerale. Hüpiknimekiri näitab sulle kõiki olemasolevaid materjale. Neist mõne kasutamiseks tuleb lihtsalt sellel klõpsata.



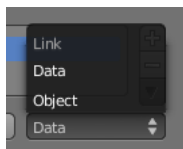
Olemasolevate
materjalide nimekirj



Filtreeritud nimekirj

- Blenderi versiooni 2.5 uuenduseks on materjalide nimekirja all olev otsinguväli. Sisestades sinna näiteks sõna "puu", filtreeritakse kõigi olemasolevate materjalide nimekirja, nii et näidatakse ainult nimes sõna "puu" kasutavaid materjale.
- 3D-vaates olles saad kiirelt linkida kõik valitud objektid [aktiivse objekti](#) materjali ning teiste aspektidega, vajutades CtrlL. See on väga kasulik, kui soovid suurt hulka objekte panna kasutama sama materjali; lihtsalt vali need ning seejärel objekt, millel juba on soovitud materjal, vajuta CtrlL ja lingi nad selle "ülemobjektiga".

Materjali linkimine objekti või objekti andmetega



Materjal objekti või
objekt andmetega
linkimine

Materjali saab linkida objekti või objekti andmetega.

Palun laiendage seda osa.

Materjali valikud

Kui objektiga on lingitud juba vähemalt üks materjal, ilmub palju uusi paneele, mille abil saad materjali varjutamist täpsemalt juhtida. Kõigi paneelide kasutamist seletatakse [järgmises osas](#).

[Elvaate paneel](#) püüab sulle näidata, millise tulemuse saab varjutajat erinevat tüüpi lihtsate geomeetriliste kujundite peal kasutades. Su nuppude akna paigutusest olenevalt võivad mõned paneelid olla kinni.

Materjali eelvaade

Mode: Kõik režiimid

Panel: *Shading/Material* kontekst → *Preview*

Kirjeldus

Eelvaate paneel tekitab aktiivsest materjalist ja selle omadustest kiire visuaaliseeringu (k.a selle varjutajad (Shaders), värviüleminekud (Ramps), peegelduse läbipaistvuse omadused (Mirror Transp) ja tekstuurid (Textures)). Eelvaade pakub mitmeid erinevaid kujundeid, mis tulevad uute varjutajate loomisel kasuks: mõnede varjutajate puhul (nagu need, mis kasutavad ülemineku (Ramp) värve või hajususvarjutajaid nagu Minnaert'i), on vaja suhteliselt keerulisi või spetsiifilisi eelvaate kujundeid, et aru saada, kas ehitatav varjutaja täidab oma ülesannet.

Valikud

Lame XY tasapind

Kasulik lamedate objektide nagu seinte, paberite jms tekstuuride ja materjalide loomisel.

Kera

Kasulik kerakujuliste objektide tekstuuride ja materjalide eelvaatamiseks, aga samuti metallide ja teiste valgust murdvate/läbipaistvate materjalide eelvaatamiseks (seda tänu oma ruudulisele taustale).

Kuup

Kasulik kuubikujuliste objektide tekstuuride ja materjalide eelvaatamiseks, aga samuti protseduuritekstuuride ehitamiseks. Kasutab ruudulist tausta.

Pärdik

Kasulik orgaaniliste või keerulistel mitteprimitiivvormidel põhinevate objektide tekstuuride ja materjalide eelvaatamiseks. Kasutab ruudulist tausta.

Karvad

Kasulik kiudjate objektide nagu rohu, karvade, sulgede ja juuste tekstuuride ja materjalide eelvaatamiseks. Kasutab ruudulist tausta.

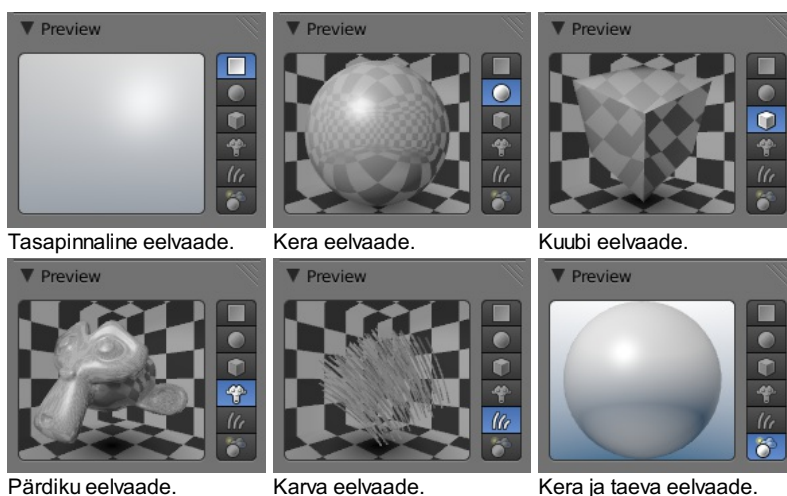
Suur kera taevaga

Kasulik kerakujuliste objektide tekstuuride ja materjalide eelvaatamiseks, kuid samuti metallide ja teiste peegeldavate materjalide ehitamiseks (tänu gradientsele taeva taustale).

Eelvaade kasutab silumist (*oversampling*)

Olgu valitud milline eelvaade tahes, kasutab see parema kvaliteedi saavutamiseks silumist. Lülita see valik välja, kui su arvuti on aeglane või vana.

Näited



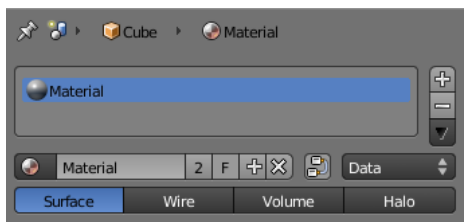
Materjalid

Mode: Kõik režiimid

Panel: *Shading/Material* kontekst → *Material*

Materjale saab objektide (ja objektandmete) külge linkida varjutuse/materjali kontekstimenüüs materjalide paneelil. Siit saad hallata, kuidas materjalid objektide/võrede/jms külge lingitakse, ning materjali selle teistes paneelides muutmiseks aktiveerida.

Kontekstimenüü

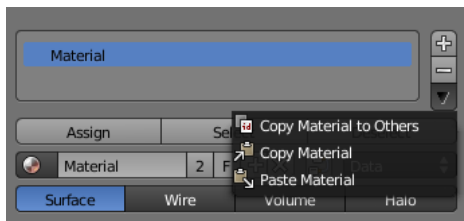


Lisatud materjal objektirežiimis

Materjali menüü ülemises ääres olev ikoonide nimekiri seletab, millises kontekstis materjali parajasti muudetakse. Ülalolevas näites on materjal Material lingitud objektiga Cube, mis on omakorda lingitud stseeniga Scene.

Vasakul oleva nõõpnõela sümbolit sisse või välja lülitades saab Blenderit käskida näidata ainult valitud materjali või muuta paneeli sisu vastavalt kontekstile.

Materjalide pesad



Kopeerimise ja kleepimise rippmenüü

Kui materjal on lingitud või loodud, saab lisada materjalipesi ning aktiveeruvad lisavalikud:

- + märk
Lisa uus materjalipesa või kopeeri valitud rida
- - märk
Eemalda valitud materjalipesa
- Nool alla
Kopeeri ja kleebi valitud materjali rida

Mitmikmaterjalid

Võredele saab lisada rohkem kui ühe materjali. Materjalid laotatakse külge (*face*) põhiselt, nagu on kirjas [mitmikmaterjalide](#) lehel. Muutmisrežiimis ilmuvad järgnevad tööriistad:

Assign (määra)
Määra valitud materjali rida valitud tippudele
Select (vali)
Valitakse valitud materjali rea tipud
Deselect (tühista valik)
Tühistab valitud materjali rea tippude valiku

Materjalide nimetamine ja linkimine

Material's name field (materjali nime väli)
klõpsa väljal ,kui tahad materjali nime muuta
Number of users (kasutajate arv; NB! numbriline väli)
Objektide või objekti andmete arv, mis seda materjali kasutab. Materjal on erinevate objektidega korraga lingitud ning selle muutmisel uuendatakse kõiki objekte. Sellele numbrile klõpsamisel tehakse 'ühe kasutajaga koopia', mis duplitseerib materjali; uus materjal on lingitud ainult aktiivse objekti/objekti andmetega.

F (*Fake user*, võltskasutaja)
Annab materjalile 'võltskasutaja', et antud materjali andmeplokk salvestataks .blend-faili ka siis, kui sel reaalsed kasutajad puuduvad.

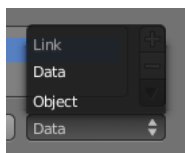
Plussmärk
Lisa uus materjal.

X-märk

Eemalda link sellele materjalile.

Nodes (sõlmed)

Määrab selle materjali materjalisõlme nuudliks ning mitte alluma materjali/värvi üleminekute/varjutajate sätetele.



Lingi materjal
objekti või objekti
andmetega

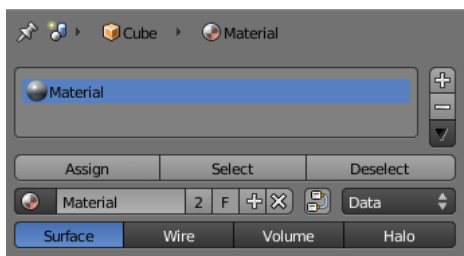
Andmehloki lingid

Linkimise hüpikmenüüs on kaks valikut: andmed (*Data*) ja objekt (*Object*). Need kaks menüüvalikut määravad, kas materjal on lingitud objekti või andmetega: võre (viimasel juhul) või kõverad või nurbid vms. Andmete menüüvalik määrab, et materjal lingitakse võre andmehloki, mis omakorda on lingitud objekti andmehloki. Objekti menüüvalik määrab, et materjal lingitakse otse objekti andmehloki.

Sellel on loomulikult tagajärjed. Näiteks võivad erinevad objektid jagada sama andmehloki. Kuna see andmehlokk määrab objekti kuju, siis iga sellele muutmisrežiimis tehtud muudatus kajastub kõigil seda kasutavatel objektidel. Lisaks jagavad kõik seda võret jagavad objektid ka selle võre andmehloki külge lingitud asju. Seega, kui materjal on lingitud võre külge, jagavad seda kõik objektid.

Teisalt aga, kui materjal on lingitud otse objekti andmehloki, siis võivad teised objektid ikkagi sama võret jagada, kuid omada teisi materjale. Lühike seletus: kui materjal on seotud objektiga, võib sul olla mitu eksemplari samadest objekti andmetest, mis kasutavad erinevaid materjale. Kui materjal on lingitud võre andmetega, ei saa sama võret kasutavad objektid erinevaid materjale kasutada.

Materjali tüüp



Muutmisrežiimis lisatud materjal

Need lülitid ütleavad Blenderile, kuhu materjal renderdusjärjestuses asetub ning milliseid materjali aspekte tuleb renderdada.

Surface (pind)

renderda objekti pinnana

Wire (traat)

renderda külgede servad traatidena (kiirejälituses seda ei toetata)

Volume (maht)

renderda objekt mahuna. Vaata [mahte](#)

Halo

renderda objekt halo osakestena. Vaata [halosid](#)

Materjali omaduste ülevaade

Materjali omaduste kõikide paneelide kasutamine on ära toodud järgmises osas.

Pind- ja traatmaterjalid

Pindmaterjalid on materjalide kõige tavalisem tüüp. Nad vastavad kindlaksmääratud pinnaga objektidele.

Traatmaterjalid muudavad lihtsalt kõik objekti servad varrasteks, mis seejärel muutuvad renderdatavateks, kuid kasutavad pindmaterjalidega samu seadeid.

Eelvaade (*Preview*)

See on valitud materjali eelvaade, mis on laotatud ühele mitmest erinevast objektist.

- Tasapind
- Kera
- Kuup
- Pärdik
- Karvad
- Suur kera koos taevaga

Hajusus (*Diffuse*)

Hajusvarjutamine simuleerib valguse langemist pinnale ning sellelt uuesti laias nurgas eemale pörkamist. Saad määrata hajusvarjutamise värvi ning selle, millist mudelit kasutatakse hajususe arvutamiseks.

Läige (*Specular*)

Läige simuleerib valgusallikate peegeldusi, mis näevad sageli välja nagu teravad heledad laigud. Saad määrata läikevarjutamise värvi ning selle, millist mudelit kasutatakse läike arvutamiseks.

Varjutamine (*Shading*)

- Emit (kiirgavus)

Lisab lisavalgustuse, otsekui materjal helendaks.

- Ambient (keskkond)

Määrab, palju globaalsest üldvalgusest materjal vastu võtab

- Translucency (läbikumavus)

Palju tagumise külje varjutusest läbi paistab. Sellega simuleeritakse pehmeid objekte nagu lehed või paber.

- Shadeless (varjutamata)

Lülitab varjutamise arvutamise välja, nii et nähtavale jääb ainult värvi informatsioon. See tekitab materjalist põhimõtteliselt "pinnavarjutaja"

- Tangent Shading (tangensvarjutus)

Kasutab tavalise varjutuse asemel materjali tangentsiaalvektoreid (puutujat); anisotroopsete varjutusefektide jaoks (nt pehme karv ja lihvitud metall). See varjutus lisandus Blenderi versioonis 2.42, vaata ka karvarenderduse sätteid allpool asuvas menüüs ning osakeste süsteemide menüüs.

- Cubic Interpolation (kuupinterpolatsioon)

Kasutab hajususe väärtuste jaoks kuupinterpolatsioone, mis annavad sujuvama ülemineku heledate ja tumedate alade vahel.

Läbipaistvus (*Transparency*)

Määrab omadused objektidele, millest valgus võib läbi paista

Peegeldus (*Mirror*)

Siit saad määrata omadused objektidele, mis peegeldavad

Pinna sisehajuvus (*Subsurface Scattering*)

Pinna sisehajuvus simuleerib pisut läbipaistvaid objekte, millesse valgus siseneb, pörkab ringi ja väljub seejärel erinevast kohast. Näiteks küünlad, inimese nahk, juust jne.

Karv (*Strand*)

Neid sätteid kasutatakse juuste ja karva materjalide renderdamiseks.

Valikud (*Options*)

- Traceable (jälitatav)

Lubab materjalil peegelduste ja murdumiste arvutamise jaoks kasutada kiirtejälitust.

- Full Oversampling (täielik silumine)

Sunnib materjali kõigi silumiskäikude jaoks täielikku varjutust ja tekstuure renderdama.

- Sky (tevas)

Renderdab materjali ilma alfata, vahetades tausta taeva vastu

- Use Mist (kasuta udu)

Kasutab selle materjali puhul udu.

- Invert Z Depth (pööra Z-sügavus tagurpidi)

Renderdab materjali küljed pahempidise Z-puhvriga.

- Z Offset (Z-nihe)

Kasutades Z-sügavuse ümberpööramist, tähistab see kõigi z-väärtuste tehisklikku nihet.

- Light Group (valgusgrupp)

Piira materjali valgusarvutused mõne kindla valgustite grupiga

- Exclusive (ainuvalgus)

Materjal kasutab ainult seda valgusgruppi

- Face Textures (küljetekstuudid)

Asendab objekti alusvärvi külgedele määratud pilttekstuuride värviga.

- Face Textures Alpha (küljetekstuudi läbipaistvus)

Asendab objekti algse läbipaistvuse külgedele määratud pilttekstuuride läbipaistvusega.

- Vertex Color Paint (tipuvärv)

Asendab objekti alusvärvi tippude värviga.

- Vertex Color Light (tipuvärvi valgus)

Lisab tippude värvi kui lisavalguse.

- Object Color (objekti värv)

Moduleeri tulemus objektipõhise värviga.

Vari (*Shadow*)

- Receive (võta vastu)

Lubab materjalil võtta vastu teiste objektide poolt heidetud varje

- Receive Transparent (võta vastu läbipaistvat)

Lubab objektil võtta vastu teiste läbipaistvate objektide poolt heidetud läbipaistvaid varje.

- Cast Only (ainult heida varju)

Sunnib selle materjaliga objekte ainult varju heitma ja ise renderdustes mitte nähtaval olema.

- Casting Alpha (heitmise läbipaistvus)

Määrab heidetava varju läbipaistvus. Kasutatakse ebaregulaarsete ja sügavarju puhvritega.

- Shadows Only (ainult varjud)

Renderdab varje kui materjali läbipaistvuse väärtusi, muutes seeläbi ainult materjali varjus olevad alad nähtavaks.

- Shadow Only Type (ainult varju tüüp)

Määrab nende varjude tüübi, mida kasutatakse, kui Ainult varjud (Shadows Only) on sisse lülitatud

- *Shadow and Distance* (vari ja kaugus)
- *Shadow Only* (ainult vari)
- *Shadows and Shading* (varjud ja varjutamine)

- Cast Buffer Shadow (heida puhvervari)

Lubab materjalil heita puhvervarjudega lampide varje.

- Buffer Bias (puhvri lävi)

Faktor, millega varjupuhvrit korrutada.

- Auto Ray Bias (automaatkiire lävi)

Ei tekita silutud normaalidega pindadel kiirtejälitusega saadud varjudes vigu

- Ray Bias (kiire lävi)

Varju kiirtejälituse lävendi väärtus, et hoida ära varju piirile tekkivad artefaktid.

- Cast Approximate (ligikaudne vari)

Luba materjalidel ligilähedase kaudvarju kasutamise puhul varju heita.

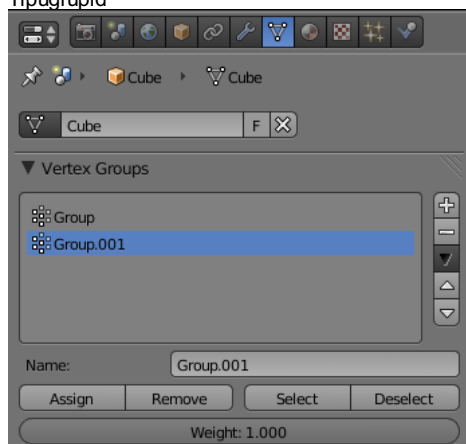
Mahtmaterjalid

Mahtmaterjalid kujutavad ennast väikeste osakestega täidetud mahte nagu pilved või suits. Nad erinevad tavamaterjalidest oluliselt, kuid neist räägitakse lähemalt [mahu lehel](#).

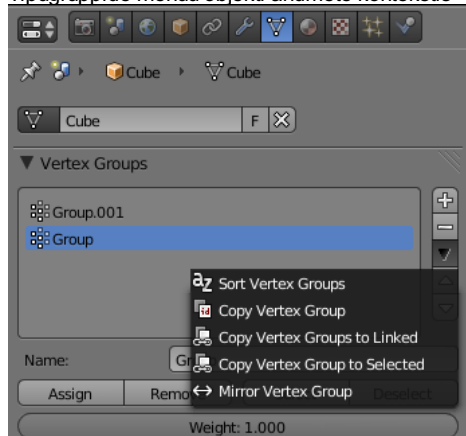
Halomaterjalid

Halomaterjalid renderdavad iga objekti punkti (*vertex*) särava täpina. See on väga kasulik osakese efektide või objektiivi peegelduse simuleerimiseks. Nendest räägitakse lähemalt [halo lehel](#).

Tipugrupid



Tipugruppide menüü objekti andmete kontekstis



Tipugruppide sorteerimise/kopeerimise rippmenüü

Materjale ei määrata mitte tipugruppidele, vaid otse tippudele. Seega on vajalik siduda tipugrupid võrega, et neile erinevaid materjale määrata, kuid samas on tipugrupid lihtne viis materjalide määramise korraldamiseks.

Tipugruppide haldamine

Uusi tipugruppe saab lihtsalt lisada, vajutades olemasolevate tipugruppide kõrval olevat + märki; nende eemaldamiseks tuleb vajutada sealsamas gruppide kõrval olevat - märki. Uued grupid lisatakse nimekirja. Tipugruppe on lihtne hallata:

1. Üles- ja allanooled liigutavad valitud tipugruppi nimekirjas üles ja alla.
2. Rippmenüüs on nimekirjas olevate tipugruppide sorteerimiseks ja kopeerimiseks veel mitu erinevat võimalust.

Vaikimisi nimetab Blender tipugrupid järjekorras Group, Group.001 jne. Iga tipugruppi saab nimevälja abil ümber nimetada.

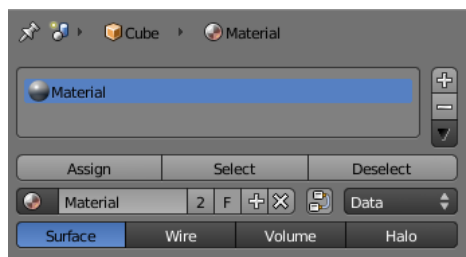
Tippude määramine tipugruppi

Tippe ei lisata tipugruppi automaatselt. Valitud tippude tipugruppi lisamiseks või nende eemaldamiseks kasuta lisamise (Assign) ja eemaldamise (Remove) nuppe.

Kui tipp lisatakse tipugruppi, määrab Blender neile kaalu väärtuse Weight — erinevatele tippudele erineva kaalu määramine on paljudel juhtudel väga kasulik, kuid materjalide puhul see mõju ei avalda.

Saad alati tipugruppe nuppude Select ja Deselect abil valida ning seda valikut tühistada.

Tippudele materjalide määramine



Materjalide menüü muutmisrežiimis

Muutmisrežiimis saad materjalide menüü kaudu määramise nupu Assign abil määrata valitud materjali valitud tippudele. Pane tähele, et see töötab kõigi teiste materjali lahtrite puhul peale esimest.

Et näha, millistele tippudele on materjal määratud, kasuta valimise (Select) ja valiku tühistamise (Deselect) nuppe.

Hajusvarjutamine (*Diffuse Shaders*)

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Hajuta (*Diffuse*)

Lihtsalt öeldes määrab hajusvarjutaja (*Diffuse Shader*) materjali üldise värvuse, kui sellele paistab valgus. Enamik reaalsust matkivaid varjutajaid annavad tulemuseks sujuva värvide ülemineku heledast tumedaks, mis ulatub kõige tugevamalt valgustatud punktist enim varjus olevate piirkondadeni; Blenderis on olemas ka mitmete eriefektide jaoks mõeldud varjutajaid.

Seaded

Kõigil hajusvarjutajatel on järgmised sätted:

Color (värv)

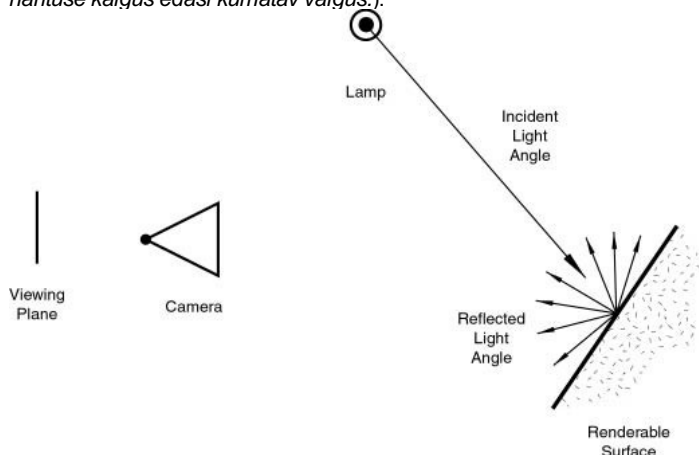
Materjali aluseks olev *hajuv värv*

Intensity (intensiivsus)

Varjutaja heledus või täpsemalt öeldes langeva valgusenergia hulk, mis reaalselt hajutatuna kaamera suunas peegeldub.

Tehnilised detailid

Pindadele langev valgus kiirgub hajususe fenomenist tulenevalt laiali, see tähendab, et seda saadetakse kõigis suundades edasi isotroopselt. See omakorda tähendab, et kaamera näeb sellelt pinnapunktilt tulevat valguse hulka oma paiknemispunktist sõltumatult üht moodi. Sellest tulenevalt ongi hajutatud valgus *vaatenurgast sõltumatu*. Loomulikult sõltub pinnale langeva valguse hulk hetke valguse langusnurgast. Kui enamus pinnale langevast valgusest peegeldatakse hajutatult, on pinnal matt välimus (vt pilti *Hajutamise nähtuse käigus edasi kumatav valgus*).



Hajutamise nähtuse käigus edasi kumatav valgus.

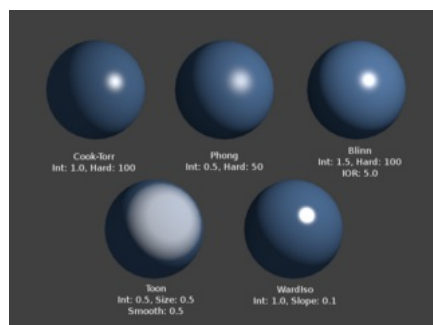
Märkus

Mõnede varjutajate nimed võivad tunduda veidrads — traditsiooniliselt nimetatakse neid inimeste järgi, kes need mudelid leiutasid.

Lambert

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



Lamberti varjutaja

See on Blenderi vaikevarjutaja ning hea universaalne tööloom.

[Johann Heinrich Lambert](#) (1728-1777)

oli Šveitsi matemaatik, füüsik ja astronoom, kes avaldas töid valguse peegeldumise kohta, millest olulisim on [seadus Beer-Lamberti seadus](#), mis sõnastab valguse neeldumise seaduse.

Seaded

Sellel varjutajal on olemas ainult vaikeparameeter, mis määrab, kui palju valgust peegeldub. Vaikeväärtus on 0.8, et võimaldada vajadusel teha heledamaid objekte.

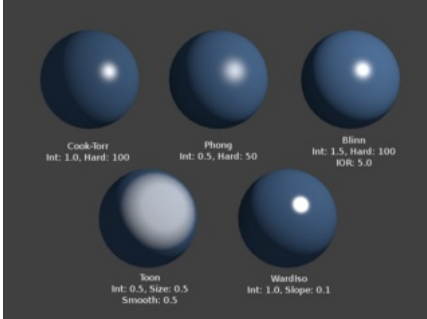


Lamberti hajusvarjutaja sätted.

Oren-Nayar

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



Oren-Nayari varjutaja

Oren-Nayar võtab hajutamise fenomenile mõnevarra 'füüsikalisema' lähenemisnurga ning arvestab ka pinna mikroskoopilise kareduse määra.

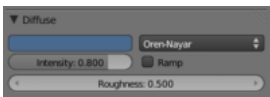
[Michael Oren](#) ja [Shree K. Nayar](#)

nende [peegelduvuse mudel](#), mis arendati välja 1990. aastate algul, on Lamberti seaduse üldistus ning seda kasutatakse arvutigraafikas laialdaselt.

Seaded

Roughness (karedus)

Pinna kareduse ja seega kiirguva hajusvalguse määr

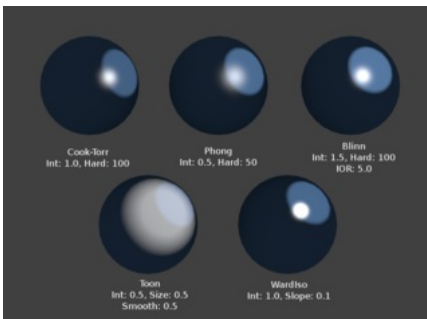


Oren-Nayari hajusvarjutaja sätted.

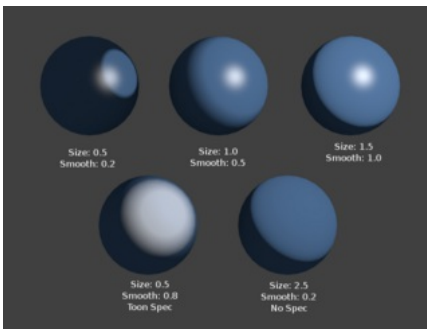
Multifilmi (*Toon*)

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



Multifilmi varjutaja (*Toon*), erinevad läiked

Multifilmi varjutaja (*Toon*) variatsioonid

Multifilmi varjutaja ei põhine füüsikal, sest see pole loodud reaalsuse matkimiseks, vaid multifilmilikku tselluloidlehtede stiili jäljendama, kus valguse ja varju vahel on selged piirid ning valgustatud/varjatud alad on ühtlased.

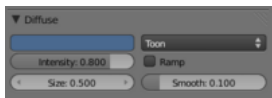
Seaded

Size (suurus)

Valgustatud ala suurus

Smooth (siledus)

Valguse ja varju piiri pehmus

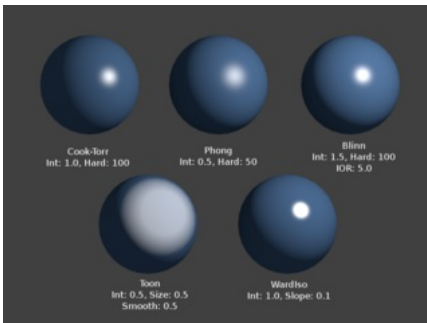


Multifilmi hajusvarjutaja sätted.

Minnaert

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



Minnaerti varjutaja

Minnaert tumendab tavalise Lamberti varjutaja osi; kui tumeduse (*Darkness*) väärtus on 1, saad täpselt sama tulemuse kui Lamberti varjutajast. Kõrgem tumeduse parameeter tumendab objekti keskk kohta (seda, millega ta vaataja poole on). Madalam tumeduse parameetri väärtus muudab objekti servad heledamaks ja annab sellele kergelt sametja välumuse.

[Marcel Minnaert](#) (1893–1970)

oli Belgia astronoom, keda huvitasid atmosfääri mõjud valgusele ja piltidele ning kes avaldas 1954. aastal raamatu pealkirjaga *Valguse ja värvi käitumine vabas õhus*.

Seaded

Darkness (tumedus)

'Valgustatud' alade tumedus (suurem väärtus) või valgusallikast eemale ulatuvate servade tumedus (madalam väärtus).

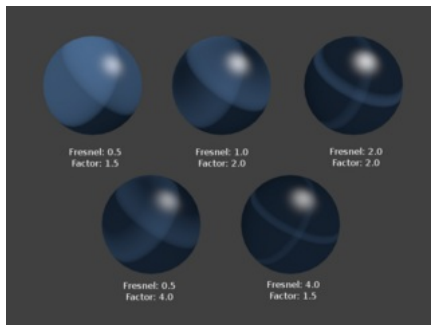


Minnaerti hajusvarjutaja seaded.

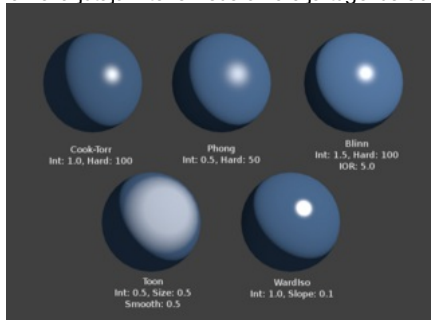
Fresnel

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



Fresneli varjutaja erinevad sätted, Cook-Torri läikevarjutaja intensiivsus on 0.5 ja tugevus 50



Fresneli varjutaja, erinevad läiked

Fresneli varjutaja puhul sõltub peegeldatud hajusvalguse hulk valguse langemisnurgast, st valgusallika suunast. Alad, mis on suunatud otse valgusallika poole, näivad tumedatena; alad, mis on valgusallikaga risti, muutuvad heledamaks.

[Augustin-Jean Fresnel](#) (1788–1827) oli Prantsuse füüsik, kes panustas oluliselt lainetel põhineva optika juurutamisse.

Seaded

Fresnel

Fresneli efekti tugevus, maksimum on 5.0.

Factor (faktor)

Sissesegatava Fresneli faktori sulatusfaktor, maksimum on 5.0.



Fresneli hajusvarjutaja seaded.

Läike varjutajad (*Specular Shaders*)

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)

Kirjeldus

Läike varjutajad loovad heledaid säravaid laike, mida näeb enamasti läikivatel pindadel ning mis jälgendavad seeläbi valgusallikate peegeldusi. Erinevalt [hajusvarjutajatest](#) sõltub läike varjutamine *vaatenurgast*. Vastavalt Snelli seadusele peegeldatakse läikivale pinnale langev valgus edasi nurga all, mis vastab valguse langemise nurgale (sõltuvalt pinnanormaalist) ning seetõttu on vaatenurk väga oluline.

Märkus

Oluline on rõhutada, et *läikepeegelduse* nähtus, millest selles peatükis räägitakse, ei ole sama mis peeglist nähtav peegeldus, vaid vastab pigem säravatele laikudele, mida läikival pinnal näeme. Kui soovid saavutada tõelisi peeglilaadseid peegeldusi, pead kasutama Blenderile sisseehitatud kiirtejälitajat. Loe selle kohta kasutusjuhendi [RENDERDAMISE](#) osast.

Seaded

Kõik läike varjutajad jagavad kahte ühist sätet:

Specular color (läike värv)

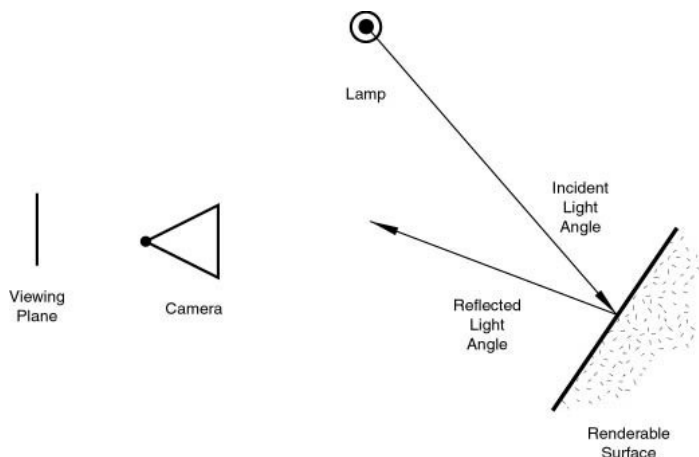
Läikelaigu värv

Intensity (intensiivsus)

Läikelaigu intensiivsus ehk heledus. Selle parameetri väärtus on vahemikus [0–1].

Tulemusena on materjalil vähemalt kaks erinevat värvi: hajususe ja läike oma. Läike värv on tavaliselt puhas valge (sama "puhas valge" nagu peegeldatud valgusallikas), kuid seda saab erinevate efektide saavutamiseks muuta (nt metallidel on värvilised läikelaigud).

Tehnilised andmed



Läikepeegeldus.

Reaalsuses tekivad nii hajusus kui ka läike peegeldus täpselt sama valguse hajumise seaduse tagajärjel. Hajusus domineerib pindadel, mis sisaldavad palju valguse lainepikkusest sõltuvaid erinevaid väikesemõõdulisi konarusi, nii et igalt pisikeselt pinnatükilt, millel on pinna kaldenurgas kas või tillukesed erinevused, peegeldatakse valgust edasi erinevates suunades.

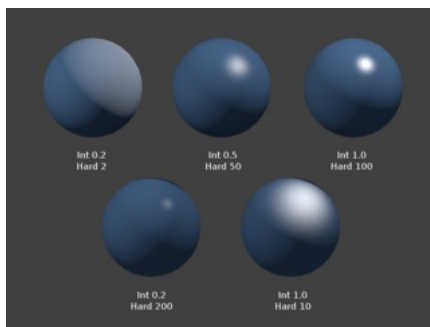
Kuid läikepeegeldus domineerib pinnal, mis on valguse lainepikkuse suhtes sile. See tähendab, et igalt pinna punktilt laiali paisatud kiired on suunatud kõik enam-vähem samas suunas ning mitte hajusalt laiali. Seega on küsimus detailide mastaabis. Kui pinnakaredus on palju väiksem kui langeva valguse lainepikkus, näib pind olevat sile ning töötab peeglina.

Kareduse mõõtkava ja valguse lainepikkuse vahetõlke mõista on keeruline: proovi ette kujutada palli (ütlemeks sentimeetrise diameetriga): kui viskad selle vastu lihvimata maakividest seina (mille konarused on detsimeetrise läbimõõduga), pörkab pall igal viskel erinevat moodi ning see kaob sul peagi ära! Kui aga viskad seda vastu betoonseina (mille konarused on umbes millimeetri suurusel), võid pörkenurka kergesti ette aimata, sest see järgib (enam-vähem!) samu reegleid nagu valguse peegeldumine.

CookTorr

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



CookTorri varjutaja (Lambert 0.8)

Kirjeldus

CookTorr (Cook-Torrance) on algeline läikevarjutaja, mis tuleb kõige rohkem kasuks läikivate plastpindade loomisel. See on natuke optimeeritum versioon Phongi varjutajast.

Robert L. Cook (Lucasfilm) ja Kenneth E. Torrance (Cornelli Ülikool)

Oma 1982. a artiklis [A Reflectance Model for Computer Graphics](#) (PDF) kirjeldasid nad "uut peegeldamise mudelit arvutiga sünteesitud piltide renderdamiseks" ning kasutasid seda mudelit metalli ja plasti simuleerimiseks.

Valikud

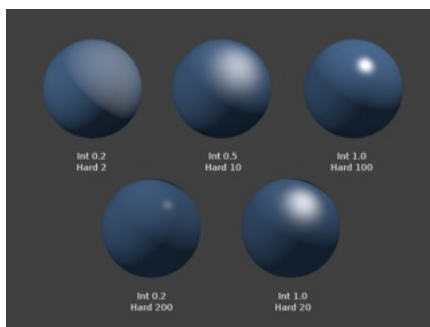
Hardness (tugevus)

Läikelaigu suurus

Phong

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



Phongi varjutaja (Lambert 0.8)

Kirjeldus

Phong on algeline varjutaja, mis sarnaneb CookTorriga, kuid on parem naha ja orgaaniliste pindade jaoks

[Bui Tuong Phong](#) (1942–1975)

oli Vietnamiis sündinud arvutigraafika pioneer, kes töötas välja esimese läikenähtuste simuleerimise algoritmi. [Tema mudel](#) sisaldas lisaks läikelaikudele ka komponente hajumise ja üldvalguse jaoks.

Valikud

Hardness (tugevus)

Läikelaigu suurus

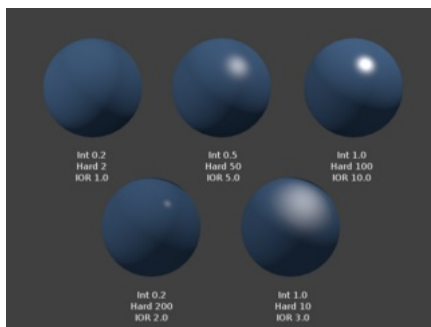
Planeedi atmosfäär

Tänu oma hägususele sobib see varjutaja hästi planeeti ümbritseva atmosfääri jäljendamiseks. Lisa planeedi ümber temast natuke suurem kera. Kasuta selle materjalina Phongi läikevarjutajat. Määra selle alfa väga madalaks (0.05), hajusus nulliks, tugevus madalaks (5), kuid läige kõrgeks (1).

Blinn

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



Blinni varjutaja (Oren-Nayari intensiivsus 0.8, karedus 0.5)

Kirjeldus

Blinn on rohkem 'füüsikaline' läikevarjutaja, mida kasutatakse sageli koos Oren-Nayari hajusvarjutajaga. Seda saab paremini kohandada, kuna lisab varem mainitud kolmele neljanda sättena *refraktsiooniindeksi* (IOR, *index of refraction*).

James F. Blinn

töötas NASA Reaktiivjõudude Laboratooriumis ning sai kuulsaks tänu tööle Carl Sagani teledokumentaalis *Kosmos* (*Cosmos*). Mudel, mida ta oma 1977. a artiklis [Models of Light Reflection for Computer Synthesized Pictures](#) (PDF) kirjeldas, sisaldas muudatusi läike intensiivsuses sõltuvalt valguse suunast ning pinnale täpsemalt paigutuvate läikelaike arvutust.

Valikud

Hardness (tugevus)

Läikelaiigu suurus. Blinni varjutajaga saab teha väiksemaid läikelaike kui Phongi või CookTorriga.

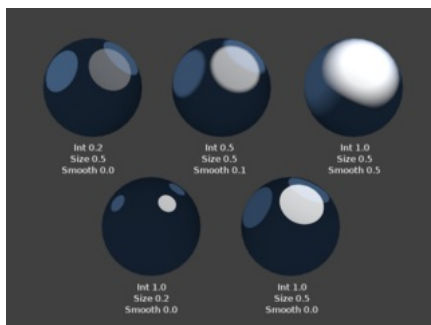
IOR

Kiirte murdumise indeks (*Index of Refraction*). Seda parameetrit ei kasutata tegelikult materjali läbivate valguskiirte murdumise arvutamiseks (selleks vajaksime kiirtejälitajat), vaid selleks, et Snelli seadust kasutades täpselt välja arvutada läikepeegelduse intensiivsus ja ulatus.

Multifilmi (Toon)

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



Multifilmi läikevarjutaja (kasutab multifilmi hajusust, mille intensiivsus on 0.8, ning suurus ja siledus on ühesugused)

Kirjeldus

Multifilmi läikevarjutaja vastab multifilmi hajususe varjutajale. Selle eesmärgiks on luua teravaid ühtlase kujuga multifilmilikke läikelaike.

Valikud

Size (suurus)

Läikelaiigu suurus

Smooth (siledus)

Läikelaiigu äärte pehmus

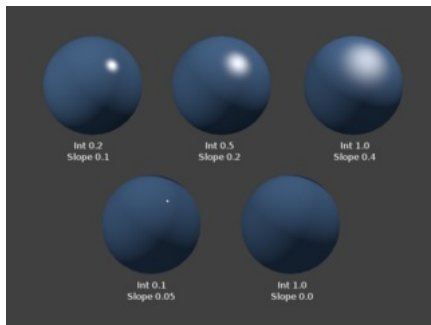
Vihje

Multifilmi varjutajat saab paremini seadistada värvüleminekuid (*Color Ramps*) kasutades.

Wardlso

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Varjutajad (*Shaders*)



Wardiso varjutaja

Kirjeldus

Wardiso on paindlik läikevarjutaja, mis on kasulik metalli ja plasti jäljendamiseks.

Gregory J. Ward

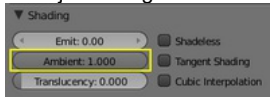
arendas välja suhteliselt lihtsa mudeli, mis vastas põhilisematele füüsikaseadustele. Oma 1992. a artiklis *Measuring and modeling anisotropic reflection* tutvustas Ward kahesuunalise peegeldumise jagunemise funktsiooni (BRDF, Bidirectional Reflectance Distribution Function), mida sestpeale arvutigraafikas laialdaselt kasutatakse, kuna selle parameetreid on lihtne määrata. Tema mudel suudab kirjeldada nii isotroopseid (valguse suunast sõltumatuid) kui ka anisotroopseid (suunast sõltuvaid) pindasid. Blenderis kutsutakse Wardi läikevarjutajat ikka veel **Wardi isotroopseks**, kuigi tegelikult on see anisotroopne. ([PDF](#))

Valikud

Slope (kaldenurk)

Standardne pinna kaldenurga hälve. See parameeter, mida tunti varem [ruutkeskmise](#) ehk rms-väärtusena, määrab põhimõtteliselt läikelaigu suuruse, kuid kasutab selleks teistsugust meetodit kui teised läikevarjutajat. Seetõttu on tulemuseks äärmiselt teravad laigud.

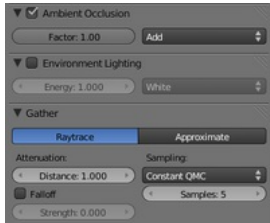
Materjali üldvalgus



Üldvalguse numbriväli
Ambient.

Iga objekti materjali seadetes (hajusvarjutuse all) on üldvalgustuse numbriväli Ambient, millega saad valida, palju üldvalgust see objekt saab.

Blenderi versioonis 2.5 on selle seade vaikeväärtus 1.0. Peaksid väärtuse seadistama vastavalt sellele, kui palju üldvalgust see objekt sinu hinnangul saab. Miski sügaval koepas ei saa üldse üldvalgust, kuid miski sissepääsu lähistel saab seda rohkem. Pea meeles, et seda efekti saad animeerida ning muuta vastavalt sellele, kuidas objekt varjudest valguse kätte liigub.



Maailma (World) kaudvarju
(Ambient Occlusion) ja
keskkonnavalgustuse
(Environment Lighting)
sätted.

Kaudvarju Ambient Occlusion ja keskkonnavalgustuse Environment Lighting sätted on leitavad maailma menüüst World ning parameetrid, mis mõjutavad mõlemat komponenti, on koondatud maailma paneeli (World) valguse koondamise osasse (Gather).

Värvüleminekud

Mode: Kõik režiimid

Panel: Kontekst Shading → alamkontekst Material → Ramps

Paljudes reaalsete maailma materjalides — nagu nahk või metallid — võib hajususe ja läikepeegelduste värv olla natuke erinev sõltuvalt valguse langemisenurgast ning sellest, palju energiat pind saab. Blenderi värvülemineku varjutaja (Ramp Shader) sätteid lasevad sul materjali jaoks valida teatud hulga toone ja määrata selle, kuidas need toonid pinna peal ühest teise üle lähevad ning kuidas nad 'päris värviga' segunevad (viimane tuleb materjalist või mõne tekstuuri väljundist).

Kirjeldus


Üleminekud lubavad sul täpselt juhtida terve materjali värvigradiendi ning mitte ainult ühtlast segamist heledaimast värvist tumedaimaks värviks või kõige rohkem valgustatud alalt kõige pimedamale alale. Lisaks erinevate seadete valgusest varju mineva gradiendi kontrollimisele on üleminekul ka 'normaalide' sisend, mis määrab gradiendi kaamerale suunatud ja kaamerast eemale suunatud pindade vahel. Seda kasutatakse sageli metallik-autovärvi sarnaste materjalide puhul, mis muudavad vaatenurgast sõltuvalt värvi.

Kuna tekstuuri arvutamised toimuvad Blenderis enne varjutamist, võib üleminekute varjutaja (Ramp Shader) tekstuuri või materjali värvi täielikult asendada. Kuid segamissätete ja alfa väärtuste abil on võimalik luua Blenderi materjalide jaoks üleminekutega veel üks varjutamise kiht.

Sätted

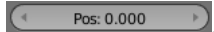


Üleminekute paneel

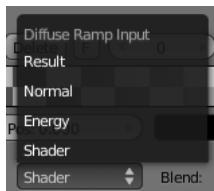
Blenderi versioonis 2.5 saab eraldi üleminekute paneeli (Ramp) sisse või välja lülitada nii hajususe (Diffuse) kui ka läike (Specular) varjutaja jaoks, kasutades nuppu .

Vaikimisi avaneb üleminekute paneel kahe värviga: esimene (tähis 0) on must ja läbipaistev (alfa=0) ning teine (tähis 1) on valge ja läbipaistmatu (alfa=1).

Värvide tähistite asukohta saab muuta kas (1) lohistades tähist värviribas (2) muutes asukoha väärtust Pos kastis



Iga tähise värvi ja alfa väärtust saab muuta, vajutades kasti .



Sisendi hüpikmenüü

Sisend (Input)

Sisendimenüüs on gradiendi määramiseks järgmised valikud:

Varjutaja (Shader)

Värvi määrab materjali varjutajast tulenev väärtus (Lambert, CookTorr). Selle puhul ei muuda värvust valguse hulk, vaid ainult valguse suund.

Energia (Energy)

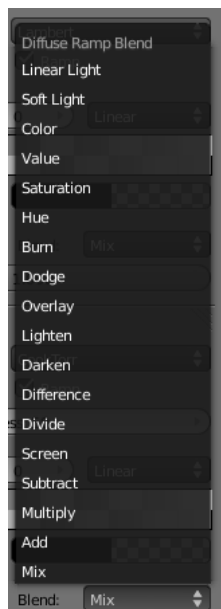
Samasugune nagu varjutaja (Shader), kuid nüüd võetakse arvesse ka valgusti energiat, värvi ja kaugust. Selle puhul muutub materjali värv, kui talle langeb rohkem valgust.

Normaal (Normal)

Värvülemineku (Ramp Shader) puhul võetakse arvesse kaameraga seotud pinna normaali. Sama tulemust on võimalik saavutada ka tekstuuriga, kuid see on siia lisatud mugavuse eesmärgil.

Tulemus (Result)

Kui kõik teised valikud töötavad iga üksiku valgusallika põhiselt, siis see valik töötab vaid kõigi varjutusarvutuste lõpus. Sellega on sul täielik kontroll terve varjutuse üle (k.a 'multifilmi' stiilis). Selles valikus alfa väärtuste muutmine on kõige lihtsam moodus anda materjalile viimane lihv.



Segamise hüpikmenüü

Segamine (Blend)

Nimekiri erinevatest segamisrežiimidest, mille abil värvüleminekut sisendist (Input) tuleva värviga segatakse.

Faktor (Factor)

See numbriväli määrab faktori, mil määral üleminek muudab sisendist (Input) tulevat värvi.


Värviribad

Mode: Kõik režiimid

Panel: Kontekst Shading → alamkontekst Material → Ramps

Värvüleminek võib sisaldada mitmest värvist koosnevat värviriba (koos alfaga), kus iga värviriba määrav värv on määratud ühe spektri punktiga. Värviribad kasutatakse nii materjalides, tekstuurides kui ka teistes kohtades, kus erinevaid toone saab samaaegselt arvutada ja kuvada.

Valikud**Add (lisa)**

Lisa värviriba keskele uus tähis vaikevärviga (neutraalne hall). Uusi tähiseid saab lisada ka hiireklahviga Ctrl LMB  gradiendi sisse vajutades, mille tulemusena lisatakse tähis vajutamise kohale, kasutades värvi, mis parajasti kursori all oli.



Delete (kustuta)

Eemaldab parajasti valitud tähise värviribast.


F

Pööra värviriba tagurpidi.

0

Aktiivse tähise number. Selle tähise väärtust kuvatakse ka paneelis ning värviriba siseselt on aktiivne tähis näidatud punktiirjoonena. Teist tähist saab valida: (1) kasutades numbriväljal  olevaid nooli, (2) vajutades tähise numbrile ning sisestades mõne värvitähise numbri või (3) vajutades klahviga LMB  värviribal olevale tähisele.

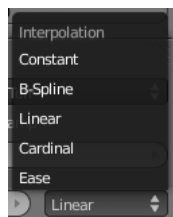
Pos (asukoht)

Aktiivse tähise asukoht värviribal (vahemikus 0.0–1.0). Värvitähiste asukohta saab muuta ka neid klahvi LMB  abil värviribal lohistades.

Värvide ümbertõstmine

Kui värvitähiseid tõstetakse ümber, siis nummerdatakse need automaatselt uuesti, et nad algaksid alati vasakul numbriga 0 ning number suureneks paremale liikudes.

Positsiooni numbriväljast (Position) paremale jääv värvinäidis (Colorswatch) kuvab aktiivse tähise värvi. Klõpsa LMB , kui tahad muuta värvivalija nähtavaks; selle kaudu saab muuta nii värvi (RGB) kui ka läbipaistvust (Alpha).



Interpolatsiooni hüpikmenüü

Linear (lineaarne)

Interpolatsiooni menüüst saab valida erinevaid tähistevahelise interpoleerimise režiime:

Ease (sujuv)

Kvadraatvalemi järgi sujuvalt tõusev/langev üleminek.

Cardinal (kardinaalne)

Kardinaalne.

Linear (lineaarne)

Lineaarne (vaikevalik). Värvidevaheline sujuvalt ühtlane üleminek.

B-Spline (bezier' kõver)

Bezier' kõver.

Constant (konstantne)

Konstantne.

Kiirtejälitusega peegeldused (*Raytraced Mirror Reflections*)

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Peegli paneel (*Mirror Transp*)

Kirjeldus

Kiirtejälitusega saab panna materjali peegeldama oma ümbrust nagu peegel. Kiirtejälitusega peegelduste põhimõte on väga lihtne: kaamerast lastakse välja kiir, mis liigub läbi stseeni, kuni põrkab vastu mõnda objekti. Kui esimene objekt, mille vastu kiir põrkab, ei ole peegeldav, siis võtab kiir endale selle objekti värvi. Kui objekt on peegeldav, siis põrkub kiir oma hetkeasukohast eemale ning liigub järgmise objektini. Ja nii edasi, kuni ta jõuab mõne mittepeegeldava objektini, mis annab kogu sellele kiirejadale oma värvi.

Lõpuks saab esimene peegeldav objekt oma peegelduvuse väärtusest (Reflectivity) sõltuvalt endale keskkonna värvid. Muidugi, juhul kui stseen koosneks ainult peegeldavatest objektidest, võiks renderdamine kesta igavesti. Seetõttu on sügavuse väärtuse Depth abil lisatud mehhanism, mis piirab ühe kiire teekonna pikkust: see väärtus määrab maksimaalse põrkumiste arvu, mida üks kiir teha saab.

Märkus

Et kasutada kiirtejälitusega peegeldusi, pead oma stseeni sätetes kiirtejälituse kõigepealt sisse lülitama. Seda saab teha stseeni/renderduse kontekstist (*Scene/Render*) → renderdamise paneelist (*Render*). Blenderi versioonist 2.37 alates on kiirtejälitus vaikimisi sisse lülitatud.

Peegeldamise paneelis olev *värvikastike* näitab tagasipeegeldatava valguse värvi. Tavaliste peeglite jaoks kasuta valget. Kuid mõned peeglid (nt metallid) toonivad peegeldust ning sellisel puhul saad sa näidisele vajutades värvi muuta. Peegelduse hulka määrab peegelduvuse väärtus Reflectivity. Kui see väärtus on suurem kui 0, siis peegelduvus aktiveeritakse ning peegeldust toonitakse vastavalt värvi näidises olevale toonile.

Seaded



Peegli paneel

Lülita kiirtejälitusega peegeldused sisse (*Enable raytraced reflections*)

lülita kiirtejälitusega peegeldused sisse või välja

Reflectivity (peegelduvus)

Määrab objekti peegelduvuse. Kasuta täiusliku peegli jaoks väärtust 1.0 ning kui sa üldse peegeldusi ei soovi, siis väärtust 0.0.



Peegli värvi valimine

Värvinäidis

Peegelduse värvus

Vaikimisi peegeldab täiuslikult peegeldav materjal nagu kroom (või peegel) oma keskkonna täpseid värve. Kuid mõned teised samavõrd peegeldavad materjalid toonivad peegeldatavat omaenda värviga. See juhtub näiteks hästi poleeritud vase või kulla puhul. Et seda Blenderis saavutada, pead sa vastavalt muutma peegli värvust (*Mirror Color*). Peegli värvuse muutmiseks vajuta lihtsalt peegli paneelis olevasse värvikastikesse ja vali uus värv.

Fresnel

Määrab Fresneli efekti suuruse. Fresneli efektist määrab, kui peegeldav materjal on, ning see sõltub vaatesuuna ja pinnanormaali vahelisest nurgast. Tavaliselt on materjal seda peegeldavam, mida suurem on see nurk (enamasti juhtub see objektide äärealadel).

Blend (segamine)

Faktor, millega saad määrata, kuidas peegeldavate ja mittepeegeldavate alade üleminek tekib.

Depth (sügavus)

Maksimaalne valguse siseste peegelduste arv. Kui su stseenis on palju peegeldavaid objekte ja/või su kaamera suurendab mõnda sellist objekti ja sa soovid peegeldatud objekti peegelduses näha seda ümbritsevaid peegeldusi (!), pead seda väärtust suurendama. Sellisel juhul on sügavuse väärtus 4 või 5 enamasti hea valik.

Max Dist (Maksimaalne kaugus)

Maksimaalne peegeldatud kiirte kaugus kaamerast (Z-sügavus) Blenderi ühikutes. Sellest kaugemale jäävad peegeldused hajuvad ja hoiavad seeläbi kokku arvutamise aega.

Fade to (hajumine)

Värvus, mille omandavad kiired, mis ei ristinud maksimaalse kauguse (Max Distance) sees. Materjali värv on parim

siseruumides toimuvate stseenide jaoks, taeva värv (maailma seadistustes) on parim välistingimustes toimuvate stseenide jaoks.



Suzanne lõbustuspargis
([.blend](#))

Gloss (siledus)

Maalimisel on läikiv pind väga sile ja särav. Lame või vähene siledus hajutab valgust ning selle tulemuseks on väga udused peegeldused. Lisaks pole ebatasased või lihvitud, kuid teralised pinnad (nagu auto värv) täiuslikud ning seetõttu peaks nende sileduse väärtus (Gloss) olema natuke madalam kui 1.0. Parempoolses näites on vasaku peegli sileduse väärtus 0.98, keskmise oma 1.0 ja parempoolse oma 0.90. Kasuta seda sätet realistliku peegelduse saavutamiseks (kas või täiesti uduse peegli jaoks). Selle väärtusega saad ka jäljendada peeglite sügavusteravust.

Amount (suurus)

Peegelduse säravus. Väärtused alla 1.0 tekitavad hajusaid ja uduseid peegeldusi ning aktiveerivad järgnevad sätted.

Threshold (lävi)

Kohanduva sümplimise lävi. Kui uus sümplimine annab (protsentuaalselt) juurde vähem kui see väärtus, siis sümplimine peatatakse. Läve tõstmine sunnib kohanduval sümplimisel rohkem käike vahele jätma, kuid tulemuseks olevad peegeldused võivad olla mürarikamad.

Samples (sämplid)

Uduse peegelduse saamiseks kasutatavad koonusekujulised sämplid. Suurem sämplite hulk annab siledama tulemuse, kuid suurem sämplite arv muudab renderdamise aeglasemaks.



Anisotroopsed
puutujapõhiselt
peegeldavad kerad, mille
anisotroopsuse väärtus on
vastavalt 0.0, 0.75, 1.0.
([.blend](#))

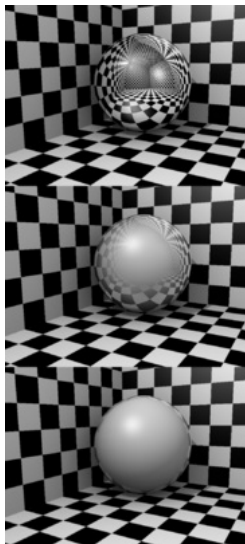
Anisotropic (anisotroopne)

Peegelduse kuju, kus 0.0 on ring ja 1.0 täielikult mööda puutujat (tangentsiaalvektorit) välja venitatud. Kui tangentsiaalvarjutus (Tangent Shading) on sisse lülitatud, renderdab Blender udused peegeldused automaatselt anisotroopsete peegeldustena.

Kui Tangent on sisse lülitatud, muudab *anisotroopse* numbrivälja anisotroopse peegelduse tugevust, mille järgi 1.0 (vaikimisi) on täiesti anisotroopne ja 0.0 täiesti ringikujuline (ehk siis sama mõjuga, kui materjali tangentsiaalvarjutus on välja lülitatud). Anisotroopsed kiirtejälitusega peegeldused kasutavad samu tangentsiaalvektoreid kui tangentsiaalvarjutus ning seetõttu saad sa nende nurka ja asetust muuta samal viisil, automaatselt genereeritud puutujatega või UV-võre koordinaatidel põhinevalt.

Näidised

Fresnel



Fresneli efekti näited,
mille puhul väärtus on
(ülalt alla) 0.0, 2.5 ja 5.0

Võtame ette väikese eksperimendi, et mõista, mida Fresneli efekt tegelikult tähendab. Mine pärast vihmast päeva õue ja seisa lombi

kõrvale. Läbi lombi näed sa maapinda. Kui sa põlvitad lombi kõrvale, paned näo maapinna lähedale ja vaatad mõnda lombi kaugemat punkti, siis veepind, mis on sulle lähemal, laseb sul näha enda alla jäävat maapinda, kuid kui pilgu kaugemale lombi teise ääre suunas liigutad, kaob maapind aeglaselt varju, kuni näed ainult taeva peegeldust. See ongi Fresneli efekt: pind, mis võib vastavalt vaatamisnurgale ja pinnanormaalile jagada nii peegeldavaid kui ka mittepeegeldavaid omadusi.

Pildil *Fresneli efekti näited, mille puhul väärtus on (ülalt alla) 0.0, 2.5 ja 5.0* demonstreeritakse seda täiuslikult peegeldava materjali puhul (Mirror Reflectivity 1.0).

Fresnel 0.0 tähistab täielikult peegeldavat materjali ning Fresnel 5.0 võiks sobida läikega materjalile. Seda on küll vaevu näha, kuid alumise pildi materjal on servades täielikult peegeldav.

Fresneli piiri sujuvust saab täpsemalt kontrollida segamise numbriväljaga Blend.

Läbipaistvus (*Transparency*)

Mode: Kõik režiimid

Panel: Varjutamise/materjali kontekst (*Shading/Material*) → Läbipaistvuse paneel (*Transparency*)

Blenderis saab materjale muuta läbipaistvaks, et valgus neid läbiks. Materjali puhul saab kasutada kolme tüüpi läbipaistvust:

Mask

Peidab tausta.

Z Transparency (Z-läbipaistvus)

Kasutab läbipaistvate külgede (*face*) puhul alfabuhvrit. Selle tüübi puhul saab muuta ainult baassätteid ning kiirte murdumist arutada ei saa.

Raytrace (kiirtejälitus)

Kasutab murdumiste arvutamiseks kiirtejälitust. Kiirtejälitusega võid saavutada keerulisi murdumisi, hajumist ja udustamist.

Kiirtejälitusega läbipaistvus (*Raytraced Transparency*)

Kiirtejälitust kasutatakse valguskiirte murdumise simuleerimiseks läbipaistvate materjalide, näiteks läätsete puhul. Kaamerast saadetakse tee kiir, mis liigub läbi stseeni, kuni jõuab mõne objektini. Kui esimene objekt, mille vastu kiir põrkab, ei ole läbipaistev, siis omandab kiir selle objekti värvuse.

Kui objekt on läbipaistev, siis jätkab kiir teekonda läbi objekti järgmise objektini ja nõnda edasi, kuni jõuab lõpuks mitteläbipaistva objektini, mis annab kogu sellele kiirejadale oma värvuse. Lõpuks saab esimene läbipaistev objekt endale oma tausta värvused, mis on proportsionaalses suhtes objekti alfa väärtusega (ning kõigi teiste kiire poolt läbitud läbipaistvate materjalide alfa väärtustega).

Aga kui kiir liigub läbi läbipaistva objekti, võib see vastavalt objekti materjali murdumisindeksile (IOR, *Index of Refraction*) oma kursilt hälbida. Kui vaatad läbi tavalise klaasist kera, paned tähele, et sellest läbi paistev taust on tagurpidi ja moondatud: see tuleneb klaasi refraktsiooniindeksist.

Kiirtejälituse sisselülitamine

Et kiirtejälitusega läbipaistvust saavutada, pead:

1. renderdamissätetes kiirtejälituse sisse lülitama. Seda saab teha renderdamiskontekstis Render → varjutamise paneelis Shading. Vaikimisi on kiirtejälitus sisse lülitatud.
2. muutma materjali alfa väärtuse millekski muuks kui 1.0.
3. et taustaks olev materjal võtaks läbi läbipaistva objekti talle langevat valgust vastu, pead tolle materjali läbipaistvuse vastuvõtmise valiku Receive Transparent sisse lülitama materjali kontekstis Material → varju paneelis Shadow.

Sätted



Läbipaistvuse paneel.

Alpha (alfa)

Määrab materjali läbipaistvuse määra.

Specular (läige)

Määrab läike värvuse alfa/hajumist.

Fresnel

Määrab Fresneli efekti tugevuse. Fresneli efektiga määratakse, kui läbipaistev on materjal; see sõltub pinnanormaali ja vaatesuuna vahelisest nurgast. Tavaliselt on materjal seda vähem läbipaistev, mida suurem see nurk on (enamasti juhtub see objektide äärealadel).

Blend (segamine)

Määrab läbipaistvate ja mitteläbipaistvate alade vahe teravust. Kasutatakse ainult siis, kui Fresneli väärtus on suurem kui 0.

IOR

Refraktsiooni- ehk murdumisindeks (*Index of Refraction*). Määrab, kui palju läbi materjali liikuvat kiirt pinnanormaali suunas murtakse ning kui palju seeläbi taustaks olevat pilti moondatakse. Vaata sektsiooni [lgapäevaste materjalide IOR-i väärtused](#) allpool.

Filter

Kiirtejälitusega saadud läbipaistvuse filtreerituse hulk. Mida kõrgem see väärtus on, seda rohkem alusmaterjali värvi on näha. Materjal jääb ikkagi läbipaistvaks, kuid materjali värv muutub järjest nähtavamaks. Vaikimisi väärtusega 0.0 välja lülitatud.

Falloff (hajumine)

Kui kiiresti valgus materjalist läbi liikumisel neeldub. Annab klaasile 'sügavust' ja 'paksust'.

Limit (piir)

Sellest paksemad materjalid ei ole läbipaistvad. Selle abil saad kontrollida läve, pärast mida hakkab filtri värvus sisse tulema.

Depth (sügavus)

Määrab kiire jaoks maksimaalse pindade läbimise arvu. Tüüpilist väärtust selle puhul ei eksisteeri. Sügavusest (Depth) väljapoole jäävad läbipaistvad objektid renderdatakse sügavmustadena, kui neid vaadatakse läbi objekti, millele see sügavuse väärtus on määratud. Teisisõnu, kui sa näed läbipaistva objekti pinnal musti alasid, siis nendest lahti saamiseks on eeldatavasti kõige lihtsam moodus selle objekti sügavuse (Depth) väärtuse suurendamine (see on kiirtejälitusega saavutatud läbipaistvate objektide puhul tavaline probleem). Vaja võib olla ka taustaobjekti materjali puhul läbipaistvate varjude sisse lülitamist.

Gloss (siledus)

Materjali läike sätted.

Amount (määr)

Murdumise selgus. Muuda see nullist väiksemaks, et saada uduseid murdumisi.

Threshold (lävi)

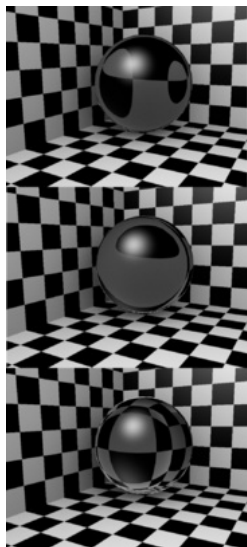
Kohaneva sümplimise lävi. Kui uus sümplimine annab (protsentuaalselt) juurde vähem kui see väärtus, siis sümplimine peatatakse.

Samples (sämplid)

Uduse murdumise saamiseks kasutatavad koonusekujulised sämplid.

Näited

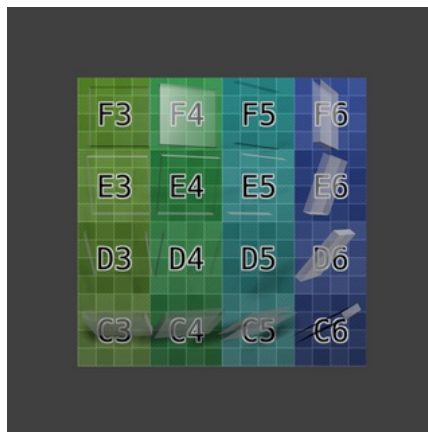
Murdumise indeks



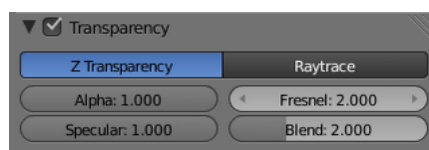
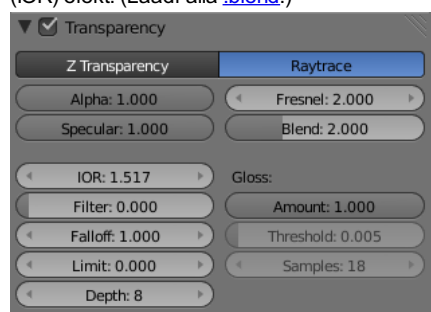
Objekti refraktsiooniindeksi (IOR) mõju tausta moonutusele: veest, klaasist ja teemandist kerad (ülalt alla).

(Objekti murdumisindeksi (IOR) mõju tausta moonutusele: veest, klaasist ja teemandist kerad (ülalt alla)). Tavalistel materjalidel on erinevad väärtused: õhk **1.000** (murdumist ei toimu), alkohol **1.329**, klaas **1.517**, plast **1.460**, vesi **1.333** ja teemant **2.417**.

Fresnel



16 klaasitükki, mis on pööratud erinevates suundades, demonstreerivad nurgast sõltuvat Fresneli efekti. Vasakul kiirtejälitusega ja paremal alfabuhvriga saavutatud läbipaistvusega. Pane tähele, et peamine erinevus on parempoolsetel puuduv murdumisindeksi (IOR) efekt. (Laadi alla [.blend](#).)



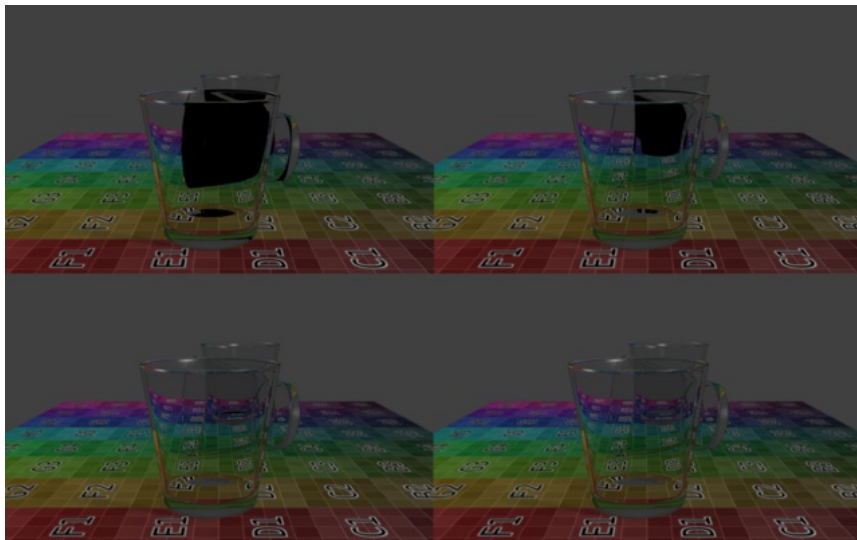
Fresneli sätted, kasutades kiirtejälitusega läbipaistvust (vasakul) ja Z-läbipaistvust (paremal).

Pane tähele läikelaiku klaastahvil kastis F4 (mille pind on suunatud pooleldi valguse ja pooleldi kaamera poole); Fresneli efekt on

nähtav reas C ja tulbas 6, kus klaasi pinnad on kaamerast eemale pööratud.

Fresneli efekti mõju saab juhtida kas segamise väärtuse Blend suurendamise või alfa väärtuse Alpha vähendamise teel.

Sügavus



Lihtne stseen kolme tasapinnale asetatud klaasi ja kolme valgustiga. Sügavuse (Depth) väärtuseks oli seatud vastavalt 4, 8, 12 ja 14, mille tulemusena oli renderdamisaeg vastavalt 24 sekundit, 34 sekundit, 6 minutit ja 11 minutit. (Laadi alla [.blend](#).)

Sügavuse (Depth) suurendamine pikendab kõvasti ka renderamisaega. Iga kord kui valguskiir läbib mõne pinna, käivitub kiirtejälituse algoritm uuesti. Ülalolevas näites on iga klaasi igal küljel nii välimine kui ka sisemine pind. Valguskiired peavad seega läbima iga klaasi puhul neli pinda.

Ja mitte ainult seda, sest iga pinna punktis võib mingi osa valgusest peegelduda tagasi või mõnda suvalisse teise suunda. Selle tulemusena peab iga punktis arvutama mitut erinevat kiirt (seda kutsutakse sageli **kiirtepuuks** [1]). Igas ülalolevas renderdatud pildis on 640×400=256 000 pikslit. Sügavuse (Depth) suurendamisel lisatakse igale pikslile vähemalt üks kiirtepuu.

Ole oma arvuti vastu hea. Objektide hoolikas paigutamine stseeni nii, et läbipaistvad objektid üksteisega ei kattu, on sageli huvitav alternatiivne lahendus.

Vihjed

Läbipaistvad varjud



Läbipaistvaid varje ei ole



Läbipaistvaid varje ei ole, keskkonnavalgustus on sisse lülitatud



Läbipaistvad varjud on sisse lülitatud, alfa väärtus on 0.0



Sama nagu eelmine, kuid alfa väärtus on 0.25



Läbipaistvad varjud, mille puhul kaudvari (ambient occlusion) on seatud korrutamise režiimi (multiply), kaugus 1 (kera raadius)



Sama nagu eelmine, kuid kaugus on 2 (kera diameeter)

Vaikimisi renderdatakse läbipaistvate objektide varjud puhta mustana justkui poleks objektid läbipaistvad. Kuid reaalsuses on enam läbipaistva objekti vari muidugi heledam.

Blenderis määratakse läbipaistvad varjud nende materjalide puhul, mille peale läbipaistva objekti vari langeb. Seda saab sisse või välja lülitada läbipaistvuse vastuvõtmise nupuga Receive Transparent materjali kontekstis Material → varju paneelis Shadow. Varju heledus sõltub seda heitva materjali alfa väärtusest Alpha.

Alternatiive läbipaistvatele kiirtejalitusega saavutatud varjudele on võimalik leida maailma kontekstist World - kaudvarju (Ambient Occlusion), keskkonnavalgustuse (Environment Lighting) ja valguse kogumise (Gather) paneelist. Samuti võib alternatiivina kasutada teksturi, mis kontrollib varju vastuvõtva materjali intensiivsuse väärtust Intensity.

Igapäevaste materjalide IOR-väärtused

Järgnevas nimekirjas on ära toodud mõned murdumisindeksi väärtused, mida saab kiirtejalitusega läbipaistvuse saavutamiseks kasutada erinevate vedelike, tahkete kehade (vääriskivid) ja gaaside puhul:

A		K		N		Š	
Agalmatoliit	1.550	Kaltsiit	1.486	Naatriumkloriid	1.544	Šampoon	1.362
Ahhaat	1.540	Kaltsiitpagu	1.486	Nailon	1.53	Šeliit	1.920
Ahhaat	1.544	Kankriniit	1.491	Natroliid	1.480	T	
Aksiniit	1.674 - 1.704	Kassiteriit	1.997	Nefeliin	1.532		Taafeiit
Aksiniit	1.675	Kaltsedon	1.544 - 1.553	Nefriit	1.600 - 1.641	Tansaniit	1.690-1.7
Akvamariin	1.575	Kivisool	1.544	Nefriit, jadeiit	1.64 - 1.667	Tantaliit	2.240
Alkohol	1.329	Klaas	1.51714		1.64 - 1.667	Teemant	2.417
Alkohol, etüül (viljapõhine)	1.36	Klaas, albiit	1.4890	O		Teflon	1.35
Aleksandriit	1.745	Klaas, kroonklaas	1.520		Obsidiaan	1.489	Teokarp
Aleksandriit	1.750	Klaas, tsink- kroonklaas	1.517	Oliviin	1.670	Teras	2.50
Almandiin	1.83	Klaas, tihe flintklaas	1.66	Oonüks	1.486	Tesiloniit	1.770
Alumiinium	1.44	Klaas, raskeim flintklaas	1.89	Opaal, must	1.440 - 1.460	Thomsoniit	1.530
Amblügoniit	1.611	Klaas, raske flintklaas	1.65548	Opaal, punane	1.430 - 1.460	Tiigrisilm	1.544
Ametüst	1.540	Klaas, lantaan- flintklaas	1.80	Opaal, valge	1.440 - 1.460	Titaniit	1.885
Ammoliit	1.600	Klaas, kerge flintklaas	1.58038	Oregoni päikesekivi	1.560 - 1.572	Topaas	1.607 - 1.627
Anataas	2.490	Klaas, keskmine flintklaas	1.62725	P		Topaas, sinine	1.610
Andalusiit	1.640	Klinohumiit	1.625 - 1.675		Padparadža	1.760 - 1.773	Topaas, oranž
Anhüidriit	1.571	Klinotsoisiit	1.724	Periklass	1.740	Topaas, roosa	1.620
Apatiit	1.632	Kloor (gaas)	1.000768	Peridoot	1.635 - 1.690	Topaas, valge	1.630
Apofülliid	1.536	Kloor (vedel)	1.385	Peristeriit	1.525	Topaas, kollane	1.620
Aragoniit	1.530	Krüsopraas	1.534	Petaliit	1.502	Tremoliit	1.600
Argoon	1.000281	Krüsopraas	1.540	Piim	1.35	Tselestiit	1.622
Asfalt	1.635	Koobalt, sinine	1.74	Plast	1.460	Tserussiit	1.804
Asuriit	1.730	Koobalt, roheline	1.97	Pleksiklaas	1.50	Tsinkiit	2.010
Atsetoon	1.36	Koobalt, violetne	1.71	Plii	2.01	Tsirkoon	1.777 - 1.987
Atsinoliit	1.618	Kolemaniit	1.586	Polüstüreen	1.55	Tsirkoon, kõrge	1.960
B		Korall	1.486 - 1.658	Prasioliit	1.540	Tsirkoon, madal	1.800
		Korall	1.658	Prehniit	1.610	Tsirkoon, teemant	2.173 - 2.21
Baritokaltsiit	1.684	Kordieriit	1.540	Pronks	1.18		1.532 - 1.554
Barüüt	1.636	Komerupiin	1.665	Prustiit	2.790		1.550
Beiniit	1.787	Korund	1.766	Purpuriit	1.840	Turmaliin	1.603 - 1.655
Benitoit	1.757	Kriit	1.510	Päevakivi, adventuriin	1.532	Turmaliin	1.624
Benseen	1.501	Kroom, roheline	2.4	Päevakivi, allbiit	1.525	Turmaliin, sinine	1.61 - 1.64
Berüll	1.57 - 1.60	Kroom, punane	2.42	Päevakivi, amasoniit	1.525	Turmaliin, kassisilm	1.61 - 1.64
Berüll, punane	1.570 - 1.598	Kroom, turmaliin	1.61 - 1.64	Päevakivi, labradoriit	1.565	Turmaliin, roheline	1.61 - 1.64
		Kroom, kollane	2.31	Päevakivi , mikrokliin	1.525	Turmaliin, paraiba	1.61 - 1.65
Berülloniit	1.553	Kroom	2.97	Päevakivi, oligoklass	1.539	Turmaliin, punane	1.61 - 1.64
Brasiliianiit	1.603	Krüsoberüll	1.745	Pärl	1.530	Tugtupiit	1.496
Brauniit	1.567	Krüsokolla	1.500	Püriit	1.810	Tärpentin	1.472
Broom (vedel)	1.661	Krokoiit	2.310	Püroop	1.740	Türkiis	1.610
D		Krüsoberüll	1.746 - 1.755	R		U	
		Kristall	2.000				Uleksiit
Danburiit	1.627 - 1.641	Kuld	0.47	Raud	1.51	Uvaroviit	1.870
Danburiit	1.633	Kumm, looduslik	1.5191	Roditsiit	1.690	V	
Diopsiid	1.680	Kupriit	2.850	Rodokrosiit	1.600		Vahtpolüstürool
Dolomiit	1.503	Kuukivi	1.518 - 1.526	Rodoniit	1.735	Vardiit	1.590
Dumortieriit	1.686			Rubiin	1.757 - 1.770	Varistsiit	1.550
E							
Eboniit	1.66						
Ekaniit	1.600						
Elavhõbe (vedel)	1.62						
Elevandiluu	1.540						
Enstatiit	1.663						
Epidoot	1.733						
Etanool	1.36						

Etüülalkohol	1.36	Kuukivi, adulaar	1.525		1.779	Vask	1.10
Euklaas	1.652	Kuukivi, albiit	1.535	Rubiin, asterism	1.76 - 1.773	Vaskoksiid	2.705
F		Kvarts	1.544 - 1.553	Rumm, valge	1.361	Vesi (0° C)	1.33346
Fenakiit	1.650	Kvartsklaas	1.45843	Rutil	2.62	Vesi (100° C)	1.31766
Fluoriit	1.434	Küaniit	1.715	S		Vesi (20° C)	1.33283
Fosgeniit	2.117	L		Safiir	1.757 - 1.779	Vesi (gaas)	1.000261
G		Labradoriit	1.560 - 1.572	Safiir, tähekujuline	1.760 - 1.773	Vesi (35° C, toatemperatuur)	1.33157
Gaseeritud joogid	1.34 - 1.356	Laminaat	1.47	Sanidiin	1.522	Vesinik (gaas)	1.000140
Geilussiit	1.517	Lasuliit	1.615	Seleen, amorfne	2.92	Vesinik (vedel)	1.0974
Glütseriin	1.473	Lasuriit, kristall	1.500	Serpentiin	1.560	Viski	1.356
Granaat, andradiit	1.88 - 1.94	Lasuriit	1.50 - 1.55	Sfaleriit	2.368	Villemiit	1.690
Granaat, demantoid	1.880 - 1.9	Leutsiit	1.509	Sideriit	1.630	Viteriit	1.532
Granaat, demantoid	1.880	Ligniit	1.660	Silikoon	4.24	Vivianiit	1.580
Granaat, grossular	1.738	Lämmastik (gaas)	1.000297	Sillimaniit	1.658	Viin	1.363
Granaat, hessoniit	1.745	Lämmastik (vedel)	1.2053	Sinhaliit	1.699	Vulveniit	2.300
Granaat, mandarin	1.790 - 1.8	M		Skapoliit	1.540	Väävel	1.960
Granaat, püroop	1.73 - 1.76	Magnesiit	1.515	Skapoliit, kollane	1.555	Õ	
Granaat, rodoliit	1.740 - 1.770	Malahhiit	1.655	Smaragd	1.560 - 1.605	Õhk	1.000
Granaat, rodoliit	1.760	Merevaht	1.530	Smaragd, kollane	1.560 - 1.605	Õli, nelgi	1.535
Granaat, spessartiit	1.810	Merevaik	1.545	Smaragd, sünteetiline	1.561	Õli, sidruni	1.481
Granaat, tsavoriit	1.739 - 1.744	Mesi, 13% veesisaldusega	1.504	Smaragd, sünteetiline	1.568	Õli, neroli	1.482
Granaat, uvaroviit	1.74 - 1.87	Mesi, 17% veesisaldusega	1.494	Smaragdiit	1.608	Õli, apelsini	1.473
H		Mesi, 21% veesisaldusega	1.484	Smithsoniit	1.621	Õli, safloori	1.466
Hambergiit	1.559	Metanool	1.329	Sodaliit	1.483	Õli, taimne (50 °C)	1.47
Hapnik (gaas)	1.000276	Metüülsalitsülaad	1.536	Spessartiit	1.79 - 1.81	Õlu	1.345
Hapnik (vedel)	1.221	Moldaviit	1.500	Spinell	1.712 - 1.717		
Hauliit	1.586	Morganiit	1.585 - 1.594	Spinell, sinine	1.712 - 1.747		
Haüiin	1.490 - 1.505			Spinell, punane	1.708 - 1.735		
Haüiin	1.502			Spodumeen	1.650		
Heelium	1.000036			Spodumeen	1.660 - 1.676		
Hematiit	2.940			Stauoliit	1.739		
Hemimorfiit	1.614			Steatiit	1.539		
Ideniit	1.655			Stihtiit	1.520		
Höbe	0.18			Strontsiumtitanaat	2.410		
Hüpersteen	1.670			Suhkruvesi 30%	1.38		
I				Suhkruvesi 80%	1.49		
Idokraas	1.713			Sünteeiline spinell	1.730		
Ioliit	1.522 - 1.578			Süsinikdioksiid (gaas)	1.000449		
J				Süsinikdisulfid	1.628		
Jadeiit	1.665			Süsiniktetrakloriid	1.460		
Jaspis	1.540						
Joodkristall	3.34						
Jõhvikamahl (25%)	1.351						
Jää	1.309						

↳

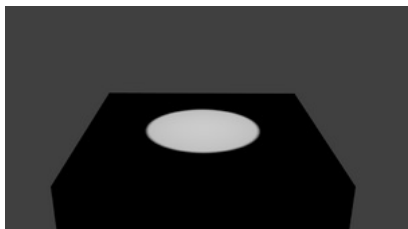
Kuidas see töötab

Reaalselt valguse teekonna arvutamine objekti pinna all on äärmiselt ebapraktiline. On näidatud, et see ei olegi vajalik, ning võimalik on kasutada teistsugust lähenemist.

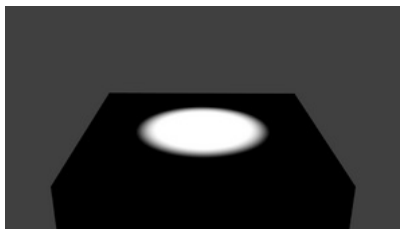
Blender arvutab sisehajumist kahe sammuga:

- Esiteks arvutatakse pinna heledus nii objekti esi- kui ka tagaküljel. See toimub peaaegu samamoodi nagu tavaline renderdamine. Arvesse võetakse nii kaudvarju, kiirgust, hajusvarjutaja tüüpi, valguse värvust jne.
- Teise sammuna renderdatakse lõpuks pilt, kuid nüüd asendab sisehajutamise varjutaja hajusvarjutaja. Valgustite asemel kasutatakse väljaarvutatud valguskaarti. Ühe pinnapunkti heledus arvutatakse seda ümbritsevate punktide heleduse "keskmisena". Vastavalt sinu valitud sätetele võidakse arvesse võtta tervet pinda ning see on tegelikult natuke keerulisem kui lihtsalt keskmise arvutamine, kuid ära vaeva ennast eriti selle taustaks oleva matemaatikaga.

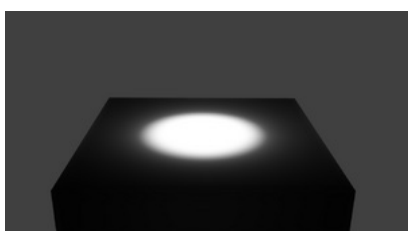
Vaatame parem, mida sisehajutamine teeb ühe konkreetse valguspunktiga.



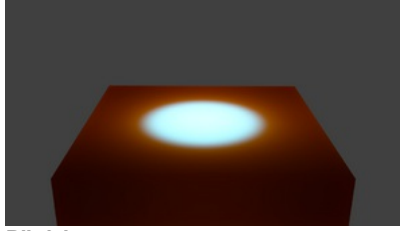
Pilt 3a: Sisehajutamist pole.



Pilt 3b: Väike sisehajumise raadius.



Pilt 3c: Suurendatud sisehajumise raadius.



Pilt 3d: Väga suure punase raadiuse väärtusega sisehajutamine.

Kui lülitad sisehajumise sisse, jagatakse valgus üle suurema pindala. Selle pindala suurus sõltub raadiuse väärtustest. Selle asemel, et jagada kõiki värve samapalju, võid iga RGB-kõvera värvi raadiuse valida eraldi.

Kui sa kasutad mõne värvuse puhul väga suurt raadiuse väärtust, jagatakse see värv ühtlaselt üle terve objekti laiali.

Pinna sisehajumise sisse lülitamine



Pilt 4: Sisehajumise paneel. Sisehajumine on juba sisse lülitatud.

- Sisehajutamise sisselülitamiseks vajuta nupule Subsurface Scattering.
- Ülemises otsas saad kasutada erinevaid eelseadistusi. Uute eelseadistuste lisamiseks või vanade eemaldamiseks vajuta + ja - nuppudele. Kui valid mõne eelseadistuse, siis määratakse raadiuste (Radius) väärtused, RGB raadius (RGB Radius) ja valguse murumise indeks (IOR) sinu eest. Ülejäänud sätteid ei määrata (sest need sõltuvad peamiselt sinu objekti suuruselt).

Pinna sisehajumise jaoks pole kiirtejälitus vajalik. Kuid kuna see sõltub langevast valgusest ja varjudest, peab su varjude arvutamine olema täpne (mis võib nõuda kiirtejälitust).

Seaded

Valguse hajumist juhivad numbrinupud:

IOR (*Index of Refraction*, Valguse murumise indeks)

Valguse murumise indeks määrab langeva valguse hajumise. Kõrgemad väärtused tähendavad, et valgus hajub kiiremini. See efekt on suhteliselt märkamatu ning muudab jagunemise funktsiooni minimaalselt. Pärast paljude materjalidega katsetamist näib, et sobilik väärtus võiks jääda **1.3** ja **1.5** vahele. Kui tead täpselt, millist materjali soovid simuleerida, [uuri meie murumisindeksi tabelit](#).

Scale (mõõtkava)

Sinu objekti mõõtmed Blenderi ühikutes, mille raames tahad hajumise efekti näha. Mõõtkava väärtus 1.0 tähendab, et **1** Blenderi ühik võrdub **1** millimeeter, mõõtkava väärtus **0.001** tähendab, et **1** Blenderi ühik võrdub **1** meeter. Kui soovid välja arvutada, millist mõõtkava oma stseenis kasutada, tee seda valemiga: (suurus Blenderi ühikutes)/(päris maailma suurus millimeetrites)=mõõtkava.



Sisehajumise värv

Värvikastike

Värvil on kaks mõju.

1. Kui mõelda sisehajumisest kui veidrast lambist, siis tähistaks värvinäidis selle valguse tooni.
2. See mõjutab ka laialihajumist — mida tumedam on värv, seda rohkem valgus hajub.

Seega, kui määrad tooniks rohelise, siis objekti valgustatud alad värvuvad roheliseks ning rohelist ennast hajutatakse väga vähe. Seetõttu näivad tumedamad alad punaste ja sinistena. Erinevat hajumist saab kompenseerida värvile suurema raadiuse andmisega.

RGB Radius (RGB-raadius)

Valguse udustumise raadius. Kui valgus liigub läbi objekti ning tuleb mõnest teisest punktist läbi pinna tagasi, on sel teekonnal olemas pikkus. Need nupud muudavad selle teekonna keskmist pikkust. Mida pikem on teekond, seda ühtlasemalt värv jaguneb.

Blend (segamine)

Color (värv)

See määrab, kui palju R, G, B valikud muudavad hajusvalguse väri ja tekstuure. Pane tähele, et isegi siis, kui selle seade väärtus on **0.0**, mõjutavad R, G, B sätted laialipaiskumise käitumist.

Texture (tekstuur)

Palju pinnatekstuuri koos varjutamisega hägustatakse.

Scattering Weight (laialihajumise mõju)

Front (esikülj)

Faktor, mille võrra esikülje hajumist suurendatakse või vähendatakse. Kui valgus siseneb läbi objekti esikülje, siis palju seda neeldub ja palju sellele juurde tuleb? (Tavaliselt **1.0** ehk **100%**).

Back (tagakülj)

Faktor, mille võrra tagakülje hajumist suurendatakse või vähendatakse. Valgus, mis tabab objekti tagant, võib sellest täielikult läbi liikuda ning väljuda objekti esiküljel. Seda juhtub enamasti peenikeste objektidega nagu käed ja kõrvad.

Error (arvutusviga)

See parameeter määrab, kui täpselt algoritm ümbritsevate punktidega arvestab. Selle jätmine väärtusele **0.05** peaks tulemuseks andma artefaktideta pildid. Väärtust saab renderdamise kiirendamiseks suurendada, kuid siis võivad tekkida vead. Selle sättimine väärtusele **1.0** on lihtne moodus, kuidas koos vigadega saada kiire eelvaade.

Kuidas luua omaenda sisehajuvusega materjali

Omaenda sisehajuvusega materjali loomiseks järgi neid lihtsaid samme:

- Määra sisehajuvuse värv omaenda soovitud toonile, mis enamasti on sama objekti domineeriva tooniga. Kui soovid kasutada värvide puhul erinevaid raadiusi, siis ära määra seda tooni liiga tumedaks.
- Määra mõõtkava faktor. Kui soovid saavutada suurt läbikumamist, siis kasuta kas väikseid objekte või suuri mõõtkava väärtusi.
- Määra raadiuste väärtused.
- Muuda esikülje (Front) ja tagakülje (Back) väärtustega heledust.

Näited

Nahk



Pinna sisehajumise mõõtkava suurendamine ([blend](#))

Loe lisaks

- [Arenduslogi: Pinna sisehajumine](#)
- [Ben Simonds Tõstame saladuseloori kolmekihiliselt sisehajumiselt Blendris \(Three Layer SSS in Blender Demystified\)](#)

Karvad (*Strands*)

Materjaliredaktori karvade osa on mõeldud spetsiaalselt karvaosakeste renderdamiseks. Kasutada saab kahte meetodit:

- **Hulktahukaliseid karvad (*Polygon strands*):** see on vaikemeetod (vanem). Karvad renderdatakse tasapinnaliste hulktahukadena. Hulktahukate arv sõltub renderdussammude sättest (Steps) objekti konteksti (Object) osakeste alamkontekstis (Particle) asuvas renderduspaneelis Render.
- **Karvelemendid (*Strand Primitive*):** karvelemendid saad aktiveerida vastava nupuga Strand render, mis asub osakesüsteemi renderduspaneelis Render. Karvakõvera ei salvestata hulktahukadena, vaid ainult juhtpunktidenä, mis teisendatakse hulktahukateks jooksvalt. Teine erinevus seisneb selles, kuidas töötab läbipaistvus. Selle asemel, et renderdada karvu olemasolevat süsteemi kasutades, sorteeritakse kõik karvasegmendid tagant ettepoole ning renderdatakse selles järjekorras.

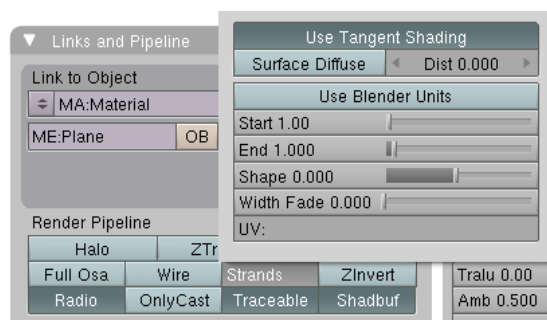
Karvaelemendid

- On mälusäästlikumad ja kiiremad ning seetõttu on võimalik renderdada suures koguses karvu ja rohtu. Paremaks toimiseks tuleks renderdussammude nupu väärtust vähendada (nt 2 peaks olema karvkatte jaoks piisav), sest tulemuseks on niikuinii sujuv kõver. Vajad 1 või 2 renderdussammu vähem kui 3D-vaateknas. Samuti vähendab suurem renderdusosade arv mälu kasutust.
- Külgedest alamobjektidega loodud karvadel on vaatekauguse vähendamise võimalus (renderduspaneelis Render alamobjekti lihtsustamise valiku Child Simplification all).
- Neid saab sujuvalt ilma lisatekstuurita otsa suunas peenemaks muuta.
- Nende puhul ei kasutata kiirtejälitust. Seega nad ei ole läbi läbipaistvate materjalide või kiirtejälitusega peeglis nähtavad (selle jaoks saad kasutada keskkonnalaotust (*Environment Mapping*)).
- Neil on probleeme kujuga, kui nende laius läheb suureks.
- Ei saa karva ulatuses kasutada UV-tekstuuri.

Hulktahukaliseid karvad

- Töötavad suurema laiusega, seetõttu saad neid kasutada tahvlite alternatiivina, sest karvade kuju saab animeerida.
- Saab kasutada karva ulatuses UV-tekstuuri.
- Kiirtejälitus näeb neid.

Karvade varjutamine



Pilt 1: Karvade varjutaja seaded.

Karvu renderdatakse sama materjaliga, mis on neid kandval küljel/tipul, kaasa arvatud varjutamine UV-tekstuuriga. Kuna sa saad igale küljele määrata enam kui ühe materjali, võib igal osakesüsteemil olla omaenda materjal ning neid kandva külje materjal võib olla erinev karvade endi omast.

Lisaks saab karvu nende pikkuse ulatuses (juurest tipuni) varjutada ühemõõtmelise tekstuuriga, kahemõõtmelist UV-tekstuuri saab kasutada ainult polügonaalsete karvade puhul.

Karvade varjutamise seaded asuvad materjali konteksti Material karvade osas Strands.

Root (juur)

Juure laius.

Tip (tipp)

Karva tipu laius.

Minimum (miinimum)

See on karvade minimaalne paksus (pikslites). Sellest suurusest allapoole jäävaid karvu ei renderdata väiksematena, vaid nende alfa keeratakse madalamaks (tegelikult töötab selline läbipaistvamaks muutmine ainult karv-elementide puhul). Sellega saavutad peene karvkatte renderdamisel parema tulemuse.

Blender Units (Blenderi ühikud)

Tavaliselt on karvad suhteliselt peened ning nende paksus on antud ekraani pikslites. Kui kasutad Blenderi ühikuid (BU), võid juure väärtuseks panna kuni 2 ühikut ja tipu väärtuseks kuni 1 ühikut. Pead arvesse võtma üldist objekti suurst, sest väikseim võimalik suurus on 0.001 ühikut. Seega, kui oled määranud, et 1 ühik tähistab 1 meetrit, siis oleks väikseim võimalik suurus 1 mm (mis on peenikese karva jaoks liiga paks).

Use Tangent Shading (kasuta tangentsiaalvarjutamist)

Arvutab valguse, nagu oleks karvad väga peenikesed ja ümarad. Seeläbi näevad nad välja heledamate ja läikivamatena. "Tangentsiaalvarjutuse" sätte väljalülitamisel renderdatakse neid ikkagi kenasti, kuid tulemuseks on tugevamad karvad, mis oleks otsekui metallist või puidust tehtud.

Shape (kuju)

Selle numbriväljaga saad kontrollida interpolatsiooni. Vaikeväärtus (0.0) tähistab lineaarset interpolatsiooni juurest (Root) tipuni (Tip). Negatiivne väärtus muudab karva peenemaks (ogaliseks), positiivne jämedamaks.

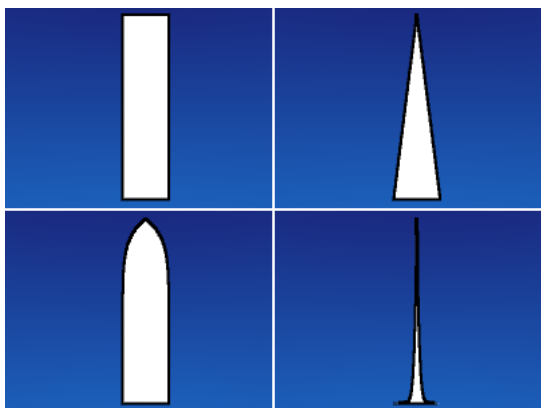


Image 2: a) Root=Tip, b) Tip=0.0, Shape=0.0, c) Shape=0.9, d) Shape=-0.9.

Width Fade (laiuse hajumine)

Sunni karvad äärtest ära hajuma. See töötab ainult primitiivkarvade puhul. 0.0-ga ei toimu üldse hajumist, 1.0 on lineaarne ärahajumine.

UV Layer (UV-kiht)

Saad hulktahukatest karvade puhul kasutada UV-tekstuuri. Pane siia UV-komplekti nimi (mitte tekstuuri oma). Pead tekstuuri laadima ka varjutamise konteksti Shading tekstuuri Texture ja materjali Material alamkontekstides (Mapping: UV; võid kasutada igat mõju sätet (Influence), mida soovid, eriti alfa väärtust, vaata *Pilt 3*).

Surface Diffuse (pinna hajusus)

Arvutab karva normaali, võttes arvesse pinnal asuvat normaali. See muudab karvade toonimist ja valgustamist kõvasti lihtsamaks, eriti primitiivkarvade puhul. Põhimõtteliselt käitub karv samamoodi nagu tavaline pind ning ei näita ülepakutud tugevaid ja suuri läikelaike.

Distance (kaugus)

Kaugus Blenderi ühikutes, millest kaugemal sulatatakse sisse pinna normaal (kui soovid kasutada pinna hajusust Surface Diffuse ainult kaugemal asuvate karvade/muru puhul).

Karvadele tekstuuri lisamine



Pilt 4: Karva alfa hajutamine...



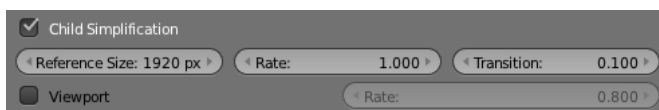
Pilt 5: ...Ja renderdamise tulemus.

Karvu saab nende pikkuse ulatuses (st juurest tipuni) tekstuurida. Selle saavutamiseks pead valima materjali alamkonteksti Material laotamise paneelis Mapping asuvast koordinaatide rippmenüüst Coordinates karva/osakese valiku Strand/Particle.

Pea kõige olulisemat seadet näitab (*Pilt 4*), mis õpetab, kuidas karva tippu alfa abil hajutada, mis annab tulemuseks kena häguse karvastiku. Tavaliselt kasutaksid sa selleks lineaarset värvisulatustekstuuri.

Loomulikult võid kasutada ka kõiki teisi omadusi, eriti värvi. Ole läikega ettevaatlik, sest karvad kipuvad kergelt särama.

Karvade renderdamise lihtsustamine



Pilt 5: Karvade renderdamise alamobjektide lihtsustamine.

Kui kasutad karvaelemente (karvade renderdamise nupp Strand render) ja oled sisse lülitanud alamobjektide interpoleerimise valiku Interpolated Children, ilmub alamobjekti lihtsustamise seade Child Simplification. Karvade renderdajal on olemas seaded, mille abil eemaldatakse karvu, kui objekti külg muutub väiksemaks.

Reference Size (alusmõõt)

See tähistab enam-vähem objekti mõõtmeid ekraanil (pikslites), pärast mida alustatakse lihtsustamist.

Rate (kiirus)

Kui kiiresti karvu eemaldatakse.

Transition (üleminek)

Ajaline üleminekuperiood, mille jooksul eemaldatavad karvad hajuvad.

Viewport (vaateaken)

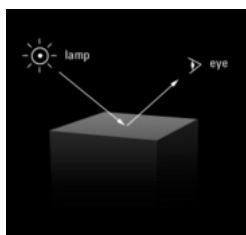
See eemaldab karvad pindadelt, mis on väljaspool vaateakent.

Rate (kiirus)

Määrab selle, kui kiiresti taolised karvad eemaldatakse.

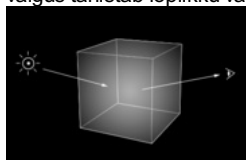
Mahu renderdamine

Mahtrenderdus on meetod, mis renderdab valgust, kui see läbib midagi, milles hajuda. Blenderi füüsikasimulaatorisse on ehitatud füüsikal põhinev mudel, mis suudab valguse käitumist erinevates keskkondades üsna realistlikult modelleerida.



Tahke renderdus

Tahke pinna renderdamise protsessis otsib kaamera mõne pinna ning arvutab seejärel välja valgusallikatest (valgusti objektidest, mitte teistest geomeetrilistest objektidest) tuleva valguse, mis objekti pinnalt kaamera suunas võiks hajuda. Kaamerasse jõudev valgus tähistab lõplikku värvi, mis ka renderdatakse.



Mahtrenderdus

Mahtude renderdamine töötab teisiti. Valgus siseneb ruumis olevasse keskkonda (mis on määratletud mahuna), milles on väikesi osakesi: suits, udu või pilved.

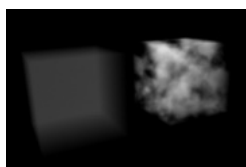
Valgus põrkab erinevate molekulide vahel ringi ning seda paisatakse laiali või see neeldub, kuni mingi osa valgusest mahust väljub ja kaamerasse jõuab. Et see maht oleks nähtav, peab renderdaja arvutama, kui suurest kogusest materjalist valgus läbi liikub ning kuidas see keskkonnas reageeris. Mahuobjekt peab olema ruumiline, näiteks suletud kinnine võre nagu kuup, ning ei tohi lihtsalt olla lame tasapind. Pildini jõudmiseks peab renderdaja sellest alast läbi liikuma ning vaatama, palju 'materiat' ala sisaldab (tihedus); sellest järeldub, kuidas valgus neeldub, hajub vms. See võib võtta aega, sest üle tuleb kontrollida palju ruumipunkte ning hinnata neist igaühe tihedust.

Mahtudele teksturi andmine

Mahule värviinfo lisamiseks kasuta [vokselandmete](#) teksturi.

Sätted

Tihedus



Ühtlane tihedus vs
tekstuuripõhine tihedus

Mahtu läbiva valgusega võib juhtuda paljutki, mis kaamerasse jõudvat lõplikku värvi mõjutavad. Need asjad kujutavad endast reaalses maailmas toimuvaid füüsilisi interaktsioone ning enamik neist sõltub keskkonna tihedusest, mis võib olla kas ühtlaselt ühesugune või varieeruv (tekstuuri poolt määratud). Seda tihedust juhtides saamegi tüüpilised 'ruumalaefektid' nagu pilved või tihe suits.

Density (tihedus)

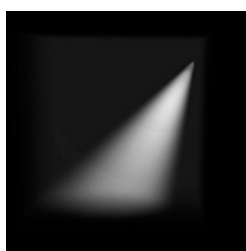
Materjali baastihedus - teistest tekstuurides pärinev tihedus liidetakse sellele

Density Scale (tiheduse määtkava)

Globaalne korutaja, mis suurendab või vähendab näilist tihedust. Seda saab kasutada selleks, et saavutada erinevates stseeni määtkavades ühtlaseid tulemusi.

Varjutamine

Hajumine



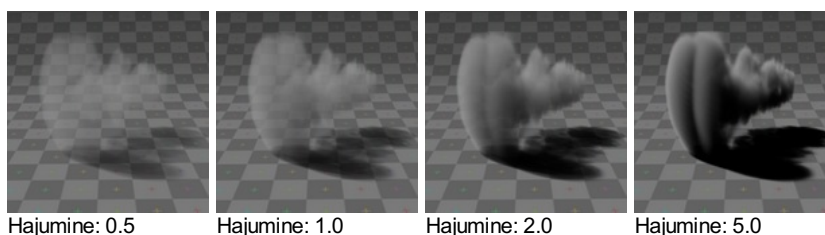
Hajuv kohtvalgus
konstantses mahus

Kui valgus siseneb välisest allikast mahtu, ei liigu see lihtsalt sellest läbi. Valgus põrkub mahus olevatelt tillukestelt osakestelt kõrvale ning mingi osa sellest valgusest jõuab kaamerasse. See omadus võimaldab näha mahust läbi liikuvaid valguskiiri, mille valgus silma suunas hajuma juhtub.

Scattering (hajumine)

Valguse hulk, mis mahust välja paisatakse. Mida rohkem valgust mahust välja hajub, seda vähem valgust jääb alles ülejäänud mahtu läbima. Selle parameetri väärtuse tõstmine võib tekitada efekti, tänu millele näib maht olevat tihedam, sest valgus paisatakse kiiresti mahu 'pinna' lähedal laiali ning mahu sisemine osa jääb seetõttu tumedamaks, kuna valgus sinna ei jõua.

Pane järgnevatel näidetes tähele, et mida vähem valgust mahust välja hajub, seda lihtsamini jõuab see läbi mahu varjuni.



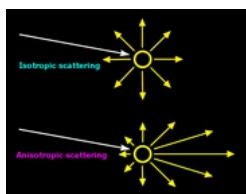
Hajumine: 0.5

Hajumine: 1.0

Hajumine: 2.0

Hajumine: 5.0

Asümmeetria



Isotroopne ja anisotroopne hajuvus

Vaikemeetod valguse hajutamiseks mahu sees on selle suunamine igas suunas võrdselt: seda tuntakse isotroopse hajuvusena. Reaalses elus võivad erineda materjalid paisata valgust laiali erinevate nurkade all: seda tuntakse anisotroopse hajuvusena. Tagasihajuvus (*Backscattering*) tähendab, et rohkem valgust hajub tagasi sissetuleku suunas ning edasihajuvus (*Forward Scattering*), et valgus eelistab hajuda valguse liikumise suunas.

Asymmetry (asümmeetria)

Asümmeetria määrab tagasi- (-1.0) ja edasihajuvuse (1.0) vahet. Vaikimisiväärtus 0.0 tekitab isotroopse hajuvuse (igas suunas võrdselt).

Kandumine

Kandumine (*Transmission*) on üldine termin, mis kirjeldab mahtu läbivat valgust.

Läbikandunud valgust mõjutavad erinevad interaktsioonid, näiteks:

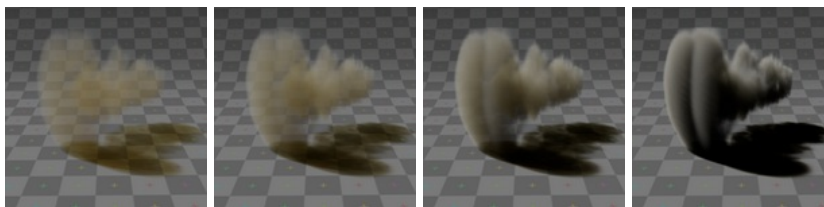
- sisenevast valgusest pärast mahust väljakiirgumist/hajumist ülejääv kogus
- pärast mahtu neeldumist (ja soojuseks muutumist) ülejääv kogus

Väljuva valguse värv määrab selle, milliseks valgus pärast mahu läbimist lõpptulemusena muutub.

Transmission Color (läbikanduv värvus)

Lõplik valguse värvus pärast mahu läbimist.

Märka allolevatel näidetes, et mida rohkem valgus mahust välja hajub, seda vähem jääb seda järele läbikanduvuseks.



Läbikanduv värv:

Läbikanduv värv:

Läbikanduv värv:

Läbikanduv värv:

kollane, hajumine: 0.5 kollane, hajumine: 1.0 kollane, hajumine: 2.0 kollane, hajumine: 5.0

Kiirgus (Emission)

Mõned mahud võivad valgust kiirata ka siis, kui seda kuskilt ei sisene, tehes seda läbi keemilise või soojusprotsesside (nagu tuli). Valgus tekib mahust endast ja on välistest allikatest saabuvas valgusest sõltumatu.

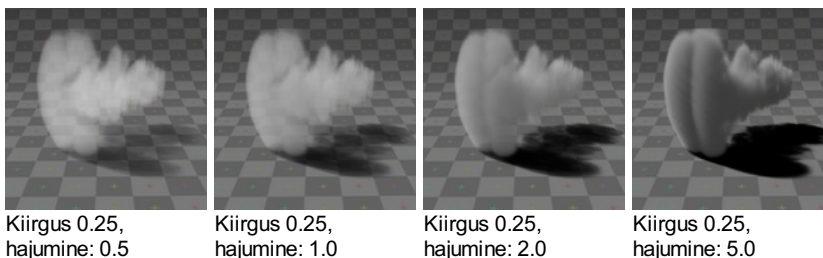
Antud hetkel ei mõjuta kiiratav valgus teisi mahte või pindu (analoogselt pindmaterjali tüübi kiirguse säte Emit).

Emission Color (kiirguse värvus)

Mahu poolt välja kiiratava valguse värvus.

Emission (kiirgus)

Kiirguse värvi intensiivsuse kordaja, mida kasutatakse mõõtkava suurendamisel ja vähendamisel.



Peegeldus

'Peegelduse' (Reflection) parameetreid saab kasutada mahust väljapaisatava valguse toonimiseks või selle heleduse muutmiseks. See mõjutab ainult valgust, mis lähtub valgustitest ja hajub, ning ei mõjuta edasikantud või kiiratud valguse värvust.

Järgnevad sätted ei ole füüsiliselt korrektsed, sest nad rikuvad energia jäävust: hajudes väljuv valgus ei mõjuta ülejäänud valgust; see kantakse läbi mahu edasi. Kui näiteks (füüsilisest seisukohast rääkides) valguse oranžid komponendid hajuvad mahust kaamera suunas välja, siis jääb alles ainult selle vastandvärv (sinine), mis jätkab mahu läbimist ning seetõttu peaks maht muutuma mitmevärviliseks (ja seega raskesti juhitavaks). Selle lihtsustamiseks on kergem lihtsalt mahu värv peegeldusparameetri abil üldise toonina ära määrata.

Reflection Color (peegelduse värvus)

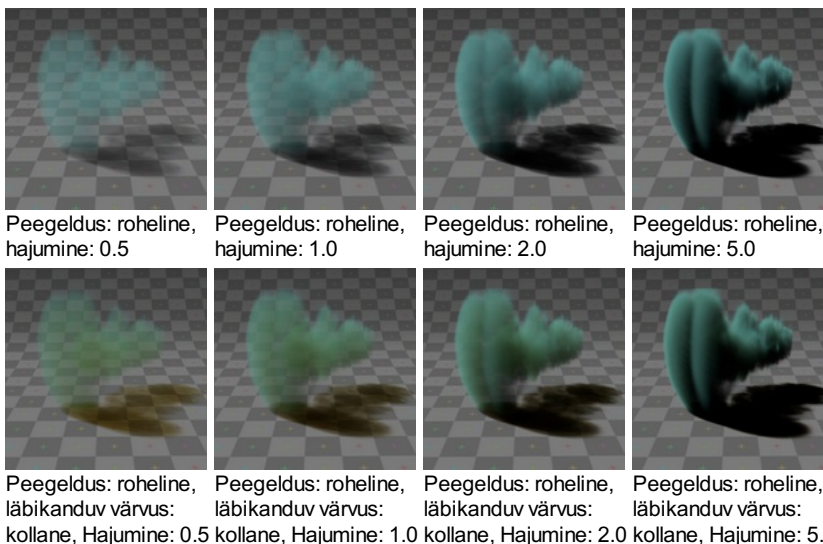
Mahust välja paisatava valguse värvus.

Reflection (peegeldus)

Peegelduse intensiivsuse korrutaja, mida kasutatakse mõõtkava suurendamisel ja vähendamisel..

Vihjed

Ideaalselt võiksid enne peegeldusparameetri kasutamist võimalikult palju teiste mahu seadistustega ära teha. Kui teed seda, mis on füüsiliselt võimalik, siis käitub materjal vastavalt ning tulemus on ettearvatavam ja seda saab kasutada paljudes erinevates valgusolukordades. Loomulikult võid sa alati ka reegleid rikkuda!



Valgustus (Lighting)

Kasutada saab mitut erinevat varjutusrežiimi, mis pakuvad palju võimalusi alates kiirelt renderduvatest kuni füüsiliselt täpseteni.

Lighting Mode (valgusrežiim)

Shadeless (varjutajata)

Varjutajata režiim on lihtsaim ning seda saab kasutada hõreda vineja udu või auru puhul.

Shadowed (varjudega)

Varjudega režiim on sarnane eelmisega, kuid kasutab ka väliste objektide varje.

Shaded (varjutatud)

Varjutatud režiim kasutab ruumilist ühekordse hajumise meetodit, mis varjutab mahtu, kui valgus sellest läbi tungib.

Multiple Scattering (mitmekordne hajumine)

Võimaldab kasutada mitut hajumisarvutust.

Shaded+Multiple Scattering (varjutatud ja mitmekordse hajumisega)

Kombineerib varjutatud ja mitmekordse hajumise režiimide omadused.

Varjutatud režiimi sätted:

External Shadows (välised varjud)

Kasuta mahust välja jäävate allikate heidetud varje (ajutine).

Light Cache (valguspuhver)

Arvuta varjutamisinfo eelnevalt vokselsõrestikuks, mis muudab varjutamise kiiremaks, kuid veidi ebatäpsemaks.

Resolution (resolutsioon)

Vokselsõrestiku resolutsioon: madalam resolutsioon on kiirem, kõrgem mälunõudlikum.

Mitmekordse hajumise sätted:

Diffusion (hajusus)

Hajutusfaktor, mis mõjutab hõgustamise tugevust.

Spread (ulatus)

Proportsionaalne vahemaa, üle mille valgus hajutatakse.

Intensity (intensiivsus)

Mitme laialipaisatud valguse energia korrutaja.

Läbipaistvus

Mask

Maskeeri taust.

Z Transparency (Z-läbipaistvus)

Kasuta läbipaistvate külgede jaoks alfabuhvrit.

Raytrace (kiirtejälitus)

Kasuta läbipaistva murdumise renderdamiseks kiirtejälitust.

Integratsioon (*Integration*)

Sammu arvutamise meetod (Step Calculation Method)

...

Randomized (juhuslik)

...

Constant (kindel)

...

Step Size (sammu pikkus)

Kaugus üksteisele järgnevate mahu tiheduse samplite vahel. Sammu pikkus määrab, kui mürarikas maht on. Kõrgemate väärtustega on renderdusaeg lühem ja müra suurem.

Depth Cutoff (sügavuse piir)

Lõpeta kiire liikumine enneaegselt, kui edasikantava valguse hulk langeb allapoole siin määratud valgustihedust; kõrgemad väärtused muudavad tihedate mahtude renderdamise kiiremaks, kuid teevad seda täpsuse arvelt.

Sätted (*Options*)

Traceable (jälitatav)

Luba sellel materjalil kasutada kiirtejälitust.

Full Oversample (täielik silumine)

Sunni seda materjali renderdama kõigi silumissammude jaoks täielikud varjutused ja tekstuurid.

Use Mist (kasuta udu)

Kasuta selle materjali puhul udu (asub Maailma valikute all).

Light Group (valgusgrupp)

Piira selle materjali valgustamine valitud grupi valgustitega.

Exclusive (ainuvalgus)

Materjal kasutab ainukesena seda valgusgruppi. Valgustid arvatakse ülejäänud stseeni valgustamisest välja.

Näited

<Need on terve wiki varjutusosa sissejuhatuse jaoks kirjutatud. See osa koosneb materjalidest ja tekstuuridest ning annab meile esialgse kirjelduse ruumilisest varjutamisest. Pane tähele, et need vastavad võreobjektidele (*mesh*). Teiste objekti tüüpide varjutamist tuleb veel uurida...>

Varjutamine on protsess ja sellele vastav kood, mille abil muutub objekt lõplikus renderdusväljundis nähtavaks. Blenderis on võreobjektide varjutamiseks neli võimalust:

1. Pind (*Surface*)
2. Ruumala (*Volumetric*)
3. Halo
4. Traat (*Wire*)

Pindvarjutus tähistab käegakatsutavat konkreetset objekti, millel on kindel (või pehme) pealiskih, näiteks tool, mõök või virsik. Pinda kirjeldatakse läbi selle hajususe, läike, peegelduse ja läbipaistvuse. Sel võib olla ka poolläbipaistev pealispind ning selle all midagi, mis valgust hajutab. Seda kutsutakse pinna sisehajumiseks. See võib olla peegelduv nagu kroom, sile plast või metall, aga ka pooleldi läbipaistev nagu klaas või vedelik.

Ruumala varjutaja peab objekti ruumi mahuks, mis on täidetud mikroskoopiliste osakestega nagu näiteks pilved, suits, udu, sudu, salapärased loitsud või aur. Kui valgus mahtu siseneb, hajutavad need osakesed ta laiali ning osa sellest valgusest jõuab silma/kaamerasse ning me näeme seda. Mahtu kirjeldatakse tiheduse mõistega. Osakesed võivad olla ühesuguse värvusega, kuid paikneda mahu sees erineva tihedusega ning seetõttu võib üldisel kujul olla tumedamaid alasid. Tihedus võib olla terves mahu ühtlane või siis ebaühtlaselt koguneda ning anda mahule äratuntava vormi. Need mikroskoopilised osakesed võivad ka ise valgust välja anda, nagu sisaldaksid nad hõõglvel söekesi, sädemeid või kannaksid endas mingit pilves peituvat energiavälja. Tihedust võib mõjutada osakeste süsteem, mille abil saab luua selgeid jugasid või kiirgusi.

Halo varjutus muudab iga objekti tipu (*vertex*) valguspunkti ning see efekt samaneb sädemete, haldjatolmu, läigatuse või näiteks ereda päikesevalguse käes oleva teemandi sädelusega. Halodega saab edasi anda ka objektiivi sisepeegelduse sarnast tulemust, mida on näha siis, kui reaalse läätsega kaamera vaatab otse eredasse valgusallikasse, näiteks päiksesse.

Traatvarjutus renderdab objekti iga serva peene joonenena, nt traatpuur või võrk. Traatrenderdus on väga kiire ning seda saab

kasutada keerukama pinna asendusmaterjalina, et vahterenderduste ajal aega kokku hoida.

Varjutamine koosneb kahest olulisest järgust: materjalist ja tekstuuridest. Värvus, mida näed, on valguse ja varjutamise tulemus ning seetõttu võiksid lugeda ka valgustamist käsitlevat osa. Blenderis on viit tüüpi objekte, mida saab varjutada: võre (*Mesh*), kõver (*Curve*), pind (*Surface*), meta (*Meta*) ja tekst (*Text*). Allolev tabel näitab, millist tüüpi varjutus iga erineva objektitüübi puhul kasutatav on. Pea meeles, et mitte-võreobjekte saab nende algsest tüübist võreks teisendada, nii et tegelikult on kõigi objektitüüpide puhul põhimõtteliselt kasutatavad kõik varjutamisvõimalused.

Vastavalt objektitüübile võimalikud varjutused

		Pind	Halo	Traat	Ruumala
Võre	jah		täielikult	jah	jah
Kõver	kui on tsükliiline või eendatud		ei	ei	
Pind	jah		ei	jah	
Meta	jah		ei	ei	
Tekst	jah		ei	ei	

Sätted

Sissejuhatus

Lisaks just äsja kirjeldatud moodustele, kuidas luua materjale materjalipaneelidel olevate sätete abil, võimaldab Blender luua materjale alusmaterjali juhtimise teel läbi sõlmevõrgu. Iga sõlm teeb materjaliga ühe operatsiooni (muutes selle väljanägemist, kui see on võrele lisatud) ning saadab tulemuse edasi järgmisele sõlmele. Sel moel on võimalik saavutada väga keerulise käitumisega materjale.

Peaksid olema juba tuttav üldiste materjali mõistetega ning sellega, kuidas materjale/tekstuure materjalide menüü abil luua. Sul peaks olema ka üldine arusaam Blenderis kasutatavatest teksuurikoordinaatide süsteemidest (nt genereeritud, UV jne). Me jätame siin vahele mitmed sõlmede võimalused, sest neist räägitakse täpsemalt järgmistes peatükkides. Iga osa jätkab sealt, kus varasemad pooleli jäid.

Esiteks ei muuda sõlmede süsteem materjalide menüüd kasutuks. Paljud omadused ja materjali sätted on ikkagi kättesaadavad ainult läbi materjalide paneeli (nt kiirtejalitusega peegeldamine (*Ray Mirror*)). Alates sõlmede Blenderisse lisamisest saame luua keerukamaid ja fantastilisemaid materjale, sest meie kontroll materjali üle on oluliselt suurenenud.

Igaks juhuks, kui sa pole (veel) kõigi mõistetega tuttav: kui lood sõlmede süsteemi (mida tuntakse ka "nuudelduse" (*noodles*) nime all), kirjeldad sa teatud sorti andmete töötlemise konveierit või torustikku, kus andmed "voolavad välja" sõlmedest, mis kirjeldavad erinevaid *allikaid*, "voolavad läbi" sõlmede, mis kujutavad endast erinevaid töötlemise ja filtreerimise etappe, ning lõpuks "voolavad sisse" sõlmedesse, mis kujutavad erinevaid väljundeid ja sihtmärke. Neid sõlmi saad üksteisega erineval moel ühendada ning muuta erinevaid "nupukesi" ehk parameetreid, mis iga sõlme käitumist juhivad. Läbi selle on sul tohutu hulk loomingulist vabadust. Ja see muutub väga kiiresti intuiitseks.

Alustame seega tavalisest materjalist.

Siin on meil standardne materjal, mille oleme lisanud kuubist võrele. Võiksime nagu varemgi lisada sellele värvi ja muuta muid sätteid ning see näeks kindlasti hea välja. Aga mis siis, kui tulemus pole päris selline, nagu sooviksime? Mis siis, kui soovime materjali loomist täpsemalt juhtida või lisada keerukust? Siinkohal tulevadki mängu sõlmed.

Sõlmekaarti saab luua [sõlmeredaktori aknas](#) töötades. Selles osas räägime:

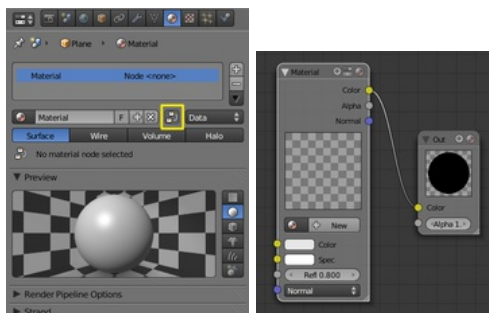
- Materjalisõlmede sisselülitamisest.
- Sõlmeredaktori aknast, selle elementaarsest juhtimisest ja sõlmedega töötamisest.
- Spetsiaalselt materjalide puhul kasutatavatest sõlmetüüpidest.

Sõlmeredaktorisse sisenemine

Sisene kõigepealt [sõlmeredaktoris](#) ja kontrolli, et seal oleks valitud materjalisõlmede nupp (keraga ikoon) ja mitte komposiitimis- või teksuurisõlmede nupp.

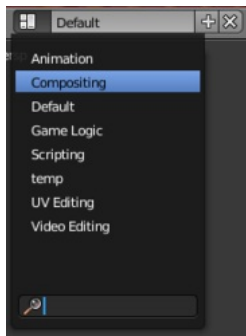
Materjali nuppude alt sõlmematerjalide sisse lülitamine

Võtame alusmaterjali ja vajutame materjali paneelis või sõlmeredaktoris selle nime kõrval olevat sõlmede nuppu Nodes. Näed materjalipaneelis muutust.



Materjali menüü koos sisselülitatud sõlmedega

Ütleksid Blenderile, et ta muudaks valitud materjali sõlmepuuks. Enamik paneele, mis tavaliselt materjali menüüs nähtaval on, on nüüd peidetud.



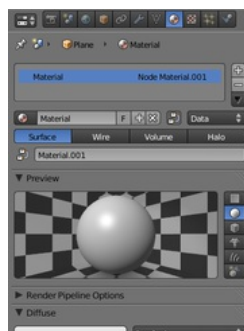
Komposiitimisekraanile (Compositing) sisenemine

Kui sisened komposiitimisekraanile (Compositing) (või vajutad kiirvalikut Ctrl←, kui sa oled vaikimisi valitud aknalaotuses), leiad ekraani ülemisest poolest sõlmeredaktori (Node Editor). Kui sa materjalisõlmed sisse lülitasid, lisati sõlmeredaktoris automaatselt materjali sõlm ja väljundi sõlm.

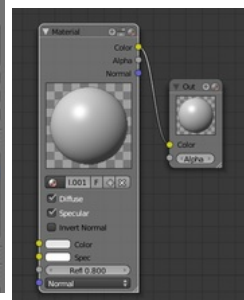
Võid ka aknalaotuses 3D-vaate poolitada ja muuta ühe akna sõlmeredaktoriaks.

Oluline on meeles pidada, et sa saad lisada uusi materjale (mida me saame materjali paneelist muuta nagu kõiki teisigi), juba loodud materjale või materjale teisest Blenderi failist ning samuti kasutada materjali, mille põhjal tegid sõlmepuu.

Pildil lisasime sõlmedektoris uue materjali (*Material.001*) ning pärast seda saame selle omadustele ligi ka materjali menüü kaudu.



Materjali menüü koos esimese sõlmedektorisse lisatud materjaliga



Nuudlitesse lisatud esimene materjal

Välised lingid

- [Blenderi materjalisõlmed](http://amrc.altervista.org) - selle Blenderi versiooni muutuste logi, millega tulid kaasa materjalisõlmed.

Sõlmeredaktor (*Node Editor*)

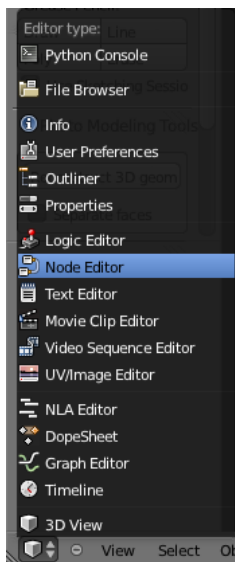
See osa seletab antud akent üldiselt, selle päisemenüüde valikuid ning kuidas sõlmede endiga ringi käia. Samuti seletab see, kuidas Blenderit sõlmi kasutama panna.

Märkus

Lisainformatsiooni selle kohta, **kus sõlmi kasutada**, loe järgmistelt lehekülgedelt:

- [Materjalide sõlmed](#)
 - [Materjalisõlmede tüübid](#)
- [Tekstuuride sõlmed](#)
 - [Tekstuurisõlmede tüübid](#)
- [Komposiitimise sõlmed](#)
 - [Komposiitimissõlmede tüübid](#)

Sisenemine sõlmeredaktorisse



Sõlmeredaktori akna valimine.

Sisene kõigepealt sõlmeredaktorisse, muutes akna tüübi sõlmeredaktori omaks (*Node Editor*). Nagu näha pildilt *Sõlmeredaktori akna valimine*, tuleb vajutada akna tüübi ikoonile ja valida hüpiknimekirjast Node Editor. Kuna sõlmekaardid võivad muutuda üpris suureks, kasuta suurt akent või loo see. Aknal on ruuduline taust ja päis.

Igas .blend-failis olevas stseenis saab olla mitu materjalisõlmede kaarti ja ÜKS komposiitimissõlmede kaart. Sõlmeredaktor näitab sõltuvalt valikust ühte neist kahest kaarditüübist.

Vihje

Soovitame lisada uue aknapaigutuse nimega "Komposiitimine" (aknapaigutuste nimekiri asub ekraani üläääres olevas infopäises), mis koosneb peamiselt ühest suurest sõlmeredaktori aknast. Minu paigutuses on nuppude aken alumises ääres ja märkmete pidamiseks külje peal tekstiredaktori aken. Kui sul on laiekraan (või isegi tavaline), võid soovi korral lisada sõlmede aknapaigutuse vasakusse serva 3D-vaate (3D View) või UV/pildi redaktori akna (UV/Image Editor): nii saad sõlmede muutmise ajal töötada ka piltide või oma mudeli kallal. Materjalisõlmede kohendamise puhul on kasulik hoida objekti kõrval lahti 3D eelrenderduse paneeli.

Nagu näha pildilt, on akna päis esmasel käivitamisel seadistusteta:



Vaikimisi sõlmeredaktori päis.

Sõlmede aktiveerimine

Materjali-, teksturi- ja komposiitimissõlmi tuleb kasutamiseks erinevalt aktiveerida. Aktiveerimise kohta vaata vastavatest peatükkidest.

Sõlmeredaktori akna tegevused

Kui kursor on aknas, saab kasutada mitut tavalist Blenderi kiirvalikut ja hiire liigutust, muuhulgas:

Kustuta

X või Del - kustutab valitud sõlme(d).

Piirdkastiga valimine

B - alustab piirdkastiga valimist. Liiguta kursor õigesse kohta, vajuta LMB ning lohista hiirt sõlmede grupi valimiseks.

Ühenduste läbilõikamine (piirdkastiga)

Vajuta **LMB**  ja lohista - alustab piirdkastiga valimist, KUID hiireklahvi vabastamisel lõigatakse läbi kõik kasti sisse jäävad ühendused.

Ennista

CtrlZ - väga kasulik, kui unustasid enne piirdkastiga valmist vajutada **B**.

Tee uuesti

CtrlY või **Ctrl⇧ShiftZ** - saad seda kasutada, kui "ennistasid" natuke liiga palju :)

Vali mitu

⇧ **Shift LMB**  või ⇧ **Shift RMB**  - vali mitu sõlme.



Haara/liiguta

G - liigutab valitud objekti.

Teosta (Execute)

E - surub sisendid läbi nuudlite (sõlmeasetuste) ja värskendab kõike.

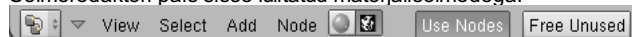
Sõlmeredaktori juhtelemendid

Sõlmede kaardid võivad muutuda üpris karvasteks (ehk siis suurteks ja keerulisteks). Akna sisu (sõlmekaarti) saab liigutada samamoodi nagu iga teise Blenderi akna oma: vajutades **MMB**  ja hiirt ringi lohistades. Hiireratta **Wheel**  üles-alla kerimine numbriklahvistiku nupud + **NumPad**/- **NumPad** suurendavad/vähendavad vaadet. Akna suurust saab muuta ning selle sisu kombineerida, kasutades tavalisi aknakäskke (vaata *Kolmemõõtmelises ruumis navigeerimine*).

Sõlmeredaktori päis



Sõlmeredaktori päis sisse lülitatud materjalisõlmedega.



Sõlmeredaktori päis sisse lülitatud komposiitmissõlmedega.

Akna päises näed järgnevaid elemente:

- **View (Vaade)** - näitab asju selgemalt;
- **Select (Vali)** - tee asju selgemalt;
- **Add (Lisa)** - sõlmede lisamiseks, organiseeritud tüübi järgi;
- **Node (Sõlm)** - valitud sõlmedega toimetamiseks;
- **Materjali-, tekstuuri- ja komposiitmissõlmede vaate valik**;
- **Sõlmede kasutamise nupp Use Nodes**
- **Kasutute sõlmede vabastamise nupp Free Unused**.

Vaate, valimise ja lisamise päisemenüüd

Nende menüüde kaudu saad teha kõige algelisemaid toiminguid:

Vaade (View)

See menüü muudab aknavaadet ning dubleerib tavalisi klaviatuuri kiirvalikuid + **NumPad** (suurenda), - **NumPad** (vähenda), **↵** **Home** (näita kõike) või nendega võrdväärseid hiireliigutusi.

Vali (Select)

Selle menüüga saad valida ühte sõlme või sõlmede gruppi ning see teeb sama, mida **A** vajutamine kõige valimiseks või piirdkastiga valimise alustamiseks **B** vajutamine.

Lisa (Add)

Siit menüüst saad sõlmi lisada. Loe järgmist peatükki, kus kirjeldatakse erinevaid sõlmetüüpe, mida saad lisada, ning seda, mida nad teevad. Sellele menüüvalikule vajutamine on sama, mis kiirvaliku ⇧ **ShiftA** vajutamine, kui kursor on aknas.

Sõlm (Node)

Selles menüüs on sõlmega töötamise tööriistad.

Tööriistad

Näita tsüklilisi sõltuvussuhteid (Show Cyclic Dependencies)

C - olgu, sa oled lisanud ja ühendanud sõlmi täpselt nagu hing ihkab ja arvuti mälu ei ole veel otsa lõppenud. Valides **Show Cyclic Dependencies**, näidatakse sulle, kus oled lõimed ringikujuliselt ühendanud. Näiteks võid lihtsalt ühendada segamise väljundi mõne teise sõlme sisendisse ning seejärel ühendada tolle sõlme väljundi tagasi segamise sisendisse, mille tulemusena on valminud ring, mida mööda piit üha edasi ja edasi jookseb. Kui see ring omaette jätta, väsis see lõpuks ära, hakkab iiveldama ja jooksub su arvuti kokku.

Peida (Hide)

H - peidab valitud sõlmed. Samamoodi nagu tipud võres.

Grupeerimine (Grouping)

Kõige tähtsam, mida selle menüüvalikuga teha saad, on iseenda poolt määratud sõlmede grupi loomine. Grupi saab seejärel

muuta ja kaardile lisada. Grupi loomiseks vali soovitud sõlmed ning kasuta menüükäsku Node → Make Group või lihtsalt klaviatuuri kiirvalikut CtrlG. Nime saad muuta väikses grupis asuvas sisestuskastis. Gruppe on lihtne ära tunda nende rohelisest päisest ja lähedatest nimedest, mida oled neile andnud.

Kustuta (Delete)

X - kustutab valitud sõlmed.

Duplitseeri (Duplicate)

⇧ ShiftD - tekitab linkimata koopias, mille seaded on samad kui originaalil.

Liiguta (Grab)

G - liigutab sõlmi vastavalt hiirele (nagu võrede puhulgi).

Duplitseeri (Duplicate) pettis sind

Uus koopia asetatakse **otse vana peale**. Kuid see ei ole ühendatud ning seetõttu ei mõjuta selle muutmine sinu pilti hoolimata sellest, et see **näeb välja** nagu oleks ühendatud, sest tema **all** olevast sõlmest tulevad välja korralikud lõimed. Et näha õigesti ühendatud sõlme, pead duplitseeritud sõlme kõigepealt eest nihutama.

Liiguta (Grab) - Väike meeldetuletus

See menüükirje ei tegelikult midagi; see on lihtsalt olemas, et tuletada sulle meelde, et võid vajutada nuppu G, kui kursor on aknas ja sa saad ka midagi reaalselt ära teha (näiteks liigutada sõlmi aknas ringi).

Tüübi valimise nupp

Sõlmed on kategooriatesse jaotatud vastavalt sellele, mida nad mõjutavad. Materjalisõlmed mõjutavad .blend-failis kasutatavaid materjale. [Materjalisõlmedega](#) töötamiseks vajuta materjali tähistavale kerale. Kui soovid töötada [komposiitmissõlmedega](#), vajuta komposiitmissõlmede kaardi vaatamiseks fotosid kujutaval nupul.

Sõlmede kasutamise päisenupp (Use Nodes)

See nupp ütleb su renderusmootorile, kas sa soovid lõpliku pildi renderdamisel kasutada materjali värvi arvutamiseks sõlmekaarti või mitte. Kui mitte, siis kaarti ignoreeritakse ning kasutatakse materjali sakkide või stseeni renderdamiseks kõige lihtsamaid seadeid.

Sõlmede lisamine ja paigutamine

Sõlmeredaktorisse lisatakse sõlmi kahel moel:

- Kasutades sõlmeredaktori tööriistaribal olevat lisamise menüüd Add ja valides sealt soovitud sõlmetüübi või
- Hoides kursorit sõlmeredaktori aknas, vajutades kiirvalikut ⇧ ShiftA ning valides hüpikmenüüst Add mõne sõlme.

Üldiselt proovi paigutada sõlmi aknas nii, et pildi arvutamine toimuks vasakult paremale, ülalt alla. Sõlme liigutamiseks vajuta mõnele selle neutraalsele alale ja lohista siis ringi. Sõlme peale võib vajutada peaaegu ükskõik kuhu ning seejärel seda ringi lohistada; ühendused proovivad ennast võimalikult hästi Bezier' kõveratena ümber kohendada.

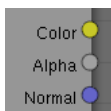
Sõlmede alla jäävad lõimed

Lõimed (kõverad, mis pesasid ühendavad) võivad liikuda mõne sõlme taha, kuid nad lihtsalt asuvad seal ning ei suhtle tolle sõlmega mitte kuidagi.

Sõlmede alla jäävad sõlmed

"Nüüd sa alles saad!" Kui sa mõnd sõlme duplitseerid, siis asetatakse uus sõlm *täpselt* vana peale. Kui sõlme sinna jätkadki (ja seda võib julgelt teha), siis sa lihtsalt ei näe, et seal on tegelikult *kaks* sõlme! Jah, sõlme vari võib olla natuke tumedam kui enne, kuid enam ei ole see visuaalselt selge, mis on millega ühendatud. Kui tekib kahtlusi, haara sõlmest kinni ja liiguta seda natuke, et näha, kas selle all miski ennast varjab.

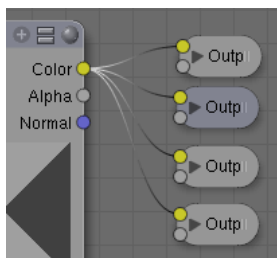
Pesad



Sõlme pesad.

Igal sõlmede aknas asuval sõlmel on "pesad" (sageli nimetatakse neid ka "ühendusteks" või sisendiks-väljundiks), mis kujutavad endast väikeseid värvilisi ringe, millesse ühendatakse sisend- ja väljundandmed (vt pilt *Sõlme pesad*).

Sõlme vasakul küljel olevad pesad tähistavad *sisendeid*, 'paremale jäävad pesad väljundeid'.



Sõlmede ühendamine.

Elu lihtsustamiseks on pesad värviga tähistatud vastavalt sellele, mis tüüpi informatsiooni nad eelistavad saata või vastu võtta. Värve on kolm:

Kollased pesad

Tähistab seda, et pesa võetakse vastu või sellest saadetakse sõlme **värvi** informatsiooni.

Hallid pesad

Tähistab väärtustega (**numbrilist**) informatsiooni. See võib olla kas üksik numbriline väärtus või "väärtuste kaart". (Väärtuste kaarti võid ette kujutada halltoonides kaardina, kus erinev heleduse/tumeduse hulk väljendab iga eri punkti väärtust.) Kui "väärtuste kaardi" pesas kasutada üksikut väärtust, kasutavad kõik kaardi punktid sama väärtust. Tavaline kasutamisiis: alfakaardid ja sõlmede numbrilise väärtusega väljad.

Sinised/lillad pesad


Tähistab **vektor-/koordinaat-/normaalide** informatsiooni.

Kui sa ei kasuta *konverterit*, millest räägime hiljem, tuleb sõlmede vahel kollased pesad ühendada kollastega, hallid hallide ja sinised sinistega.

Värvi kõrval näed sõlmes vastava pesa nime. Kuigi see ei ole alati nii, võib nime pidada vihjeks selle kohta, mis laadi infot pesa eeldab. Kuid see ei tähenda, et info *peab* seda tüüpi olema. Ma võin näiteks ühendada alfata sisaldava pesa nimega (Alpha) halli värvi pesa mõne materjalisõlme halli peegelduvuse (Reflection) pesa ning saavutada ikkagi tulemuse. Kõige olulisem on selle puhul, et tegemist on "hall-hall" ühendusega.

On erandeid, mille puhul saad ilma konverteriteta segada kollast (*nt* värviline pilt) ja halli (*nt* halltoonid). Tavaliselt lisab Blender ise konverteri, kui seda vaja on, seega eksperimenteeri vabalt. Saad kasutada vaataja väljundisõlmi "Viewer", mida seletame hiljem, nägemaks, kas ja kuidas su eksperiment töötab.

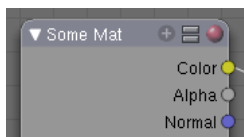
Pesade ühendamine ja lahtiühendamine

Pesade ühendamiseks vajuta soovitud pesale klahviga LMB , hoia seda all ning lohista lõim mõne teise pesani. Sobiva pesani jõudes võid hiire lahti lasta. Samuti saad pesasid ühendada, tähistades neist kaks ning kasutades kiirvalikut F. Sõlmede uuesti ühendamiseks kasuta kiirvalikut CtrlF.

Pesadevahelise ühenduse katkestamiseks hoia all Ctrl LMB  ja lohista hiirt üle ühenduse.

Väljundpesadest võib tõmmata mitu lõime ning ühendada need erinevate sõlmedega (vt pilti *Sõlmede ühendamine*). Sellisel juhul saadetakse mööda igat lõime väljundi koopiat. Sisendpesa saab ühendada aga ainult lõime.

Sõlme juhtelemendid



Sõlme ülaserv.

Sõlme ülaservas on kuni neli sõlme kontrolleri (vt pilti *Sõlme ülaserv*). Nendele vajutamine määrab selle, palju informatsiooni sõlm näitab.

Nool

Vasakul olev nool võib sõlme täielikult kokku (vt pilti *Võltimise nool*).

Plussmärk (+)

See ikoon lükkab kokku kõik pesad, mis ei ole lõimedega ühendatud (vt pilti *Plussmärk*).

Kaks ruutu (=) või "Võrdusmärk"

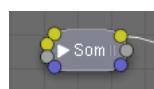
Kahe ruuduga ikoon lükkab kokku kõik sõlmes olevad väljad, mis sisaldavad kaste, milles on info (vt pilti *Menüü kokkuluukamine*).

Kera'

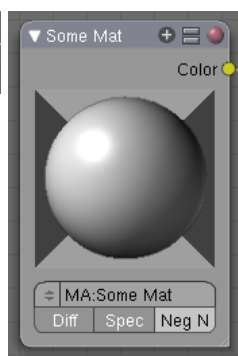
Kera ikoon võib kinni vaateakna (kui see sõlmel olemas on) (vt pilti *Kera*).

Kui kera on **punane**, võib selleks olla kolm põhjust:

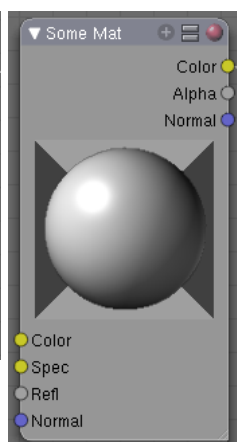
- See on [komposiitoris](#) oleva komposiitimisõlme Composite ainuke kasutatav väljund.
- See on ainuke [materjali](#) väljundisõlm Output (esimene, mis lisati).
- See on [materjali sisendisõlm](#), millele on määratud materjal (MA:).



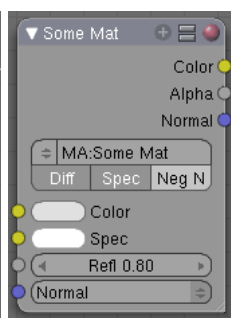
Kokkuvoltimise nool.



Plussmärk.



Menüü voltimine.

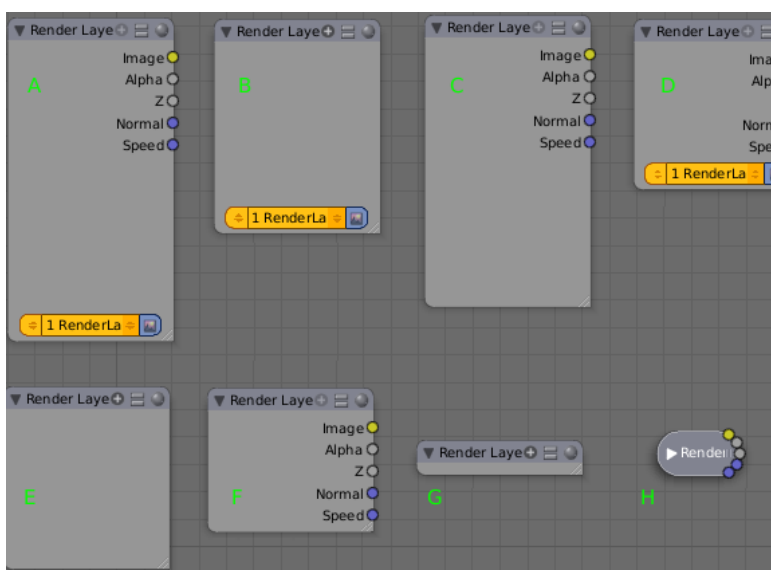


Kera.



Üheskoos.


Viimast kolme võib kasutada omavahel erinevates kombinatsioonides. Noolt, mis lükkab kokku terve sõlme, saab kasutada ainult koos plussmärgiga (vt pilti *Üheskoos*).



Sõlme ülemises ääres olevad suuruse muutmise kontrollid

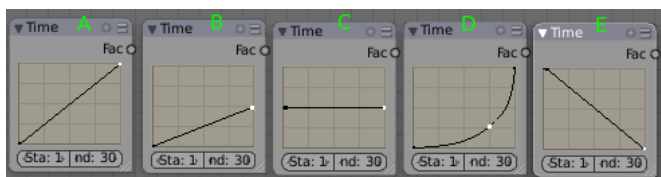
- A) Tavaline, B) Vajutatud on + märki, C) Vajutatud on = märki, D) Vajutatud on kera, E) Vajutatud on + ja = märke, F) Vajutatud on = märki ja kera G) Vajutatud on kõiki kolme H) Vajutatud on noolt.

Sõlme suuruse muutmine


Täpsemalt saab sõlme suurust mingil määral muuta, vajutades selle alumisel paremal nurgal (kus on väikesed kaldjooned) klahvi LMB  ja lohistades hiirt.

Sõlmekõverad

Mõnedel sõlmedel on kõveratega piirkonnad, mis tõlgivad sisendi väärtuse väljundi väärtuseks. Seda kõverat saab muuta, vajutades mõnele kontrollpunktile ja seda liigutades või kontrollpunkte lisades. Mõned näited on toodud allpool:



Sõlmekõvera muutmine.

Iga kõver on alguses sirge joon tõusuga 1. Sellisel kõveral on mõlemas otsas kaks väikest musta värvi juhtpunkti. Kui vajutada juhtpunkti peal LMB , siis see tähistatakse ja muutub valgeks.

Kõvera muutmine määrab, kuidas väljundit genereeritakse. Sisend, X, liigub tavaliselt lineaarselt (korrapäraste vahemaadega) mööda **alumist** telge. Liigu üles kõverani ning sealt edasi **paremale**: see määrab vastava X-i väljundi Y. Seega teise näite puhul, kus X liigub mööda alumist serva punktist 0 punktini 1.0, varieerub Y-i väärtus 0.0 ja 0.5 vahel. Kolmandas, kus X liigub mööda alumist serva punktist 0.0 punktini 1.0, jääb Y-i väärtuseks pidevalt 0.5. Seega on ülemisel pildil olevatel kõveratel mõju ajale järgmine: **A** ei

mõjuta, **B** aeglustab, **C** peatab, **D** kiirendab ja **E** pöörab aja ringi.

"Kõverate" vidin on Blenderi kasutajaliidesesse sisse ehitatud ning seda saab igal pool kasutada tingimusel, et vidinale antakse vajalikud kõvera andmed. Praegu kasutatakse seda sõlmeredaktoris ja UV aknas.


See vidin kaardistab sisendi väärtuse horisontaalselt ning tekitab kõvera kõrgusest sõltuva uue väärtuse.

Märkus: see, et igal pildil on üks kõvera punkt "valge", *ei oma* tähtsust: see tähistab lihtsalt seda punkti, mille seda õpetust ettevalmistanud autor viimati valis. Antud juhul on oluline *kõvera* kuju, mitte selle määramiseks kasutatud kontrollpunktide asukoht (või nende värv).




RGB-kõverad

Ühes kõveravidinas saab muuta mitut erinevat kõverat. Selle tavaliseks näiteks on RGB-kõver, mille esimeseks kõveraks on "kombineeritud" tulemus või "värv" ("C") ning milles on olemas ka eraldi kõverad iga RGB komponendi (punane-roheline-sinine) muutmiseks. Kõik neli kõverat on korraga aktiivsed, kuid esimesena kasutatakse arvutamiseks "C" kõverat.

Kõverapunktide valimine


- LMB  valib alati ühe punkti ja tühistab ülejäänute valiku.
- Hoia vajutamise ajal all nuppu ⇧ Shift, et valitud punktide arvu suurendada või vähendada.

Kõverate muutmine

- LMB  vajutamine ja punkti lohistamine liigutab punkte.
- Kõvera peal LMB  vajutamine lisab uue punkti.
- Ühe punkti täpselt teise peale lohistamine liidab nad üheks punktiks.
- Lohistamise ajal ⇧ Shift all hoidmisel nakkuvad punktid ruudustikuga.
- Ctrl LMB  lisab punkti.
- Valitud punktide kustutamiseks vajuta ikooni X.

Vaate muutmine

Vaikimisi vaade keskendub 0.0–1.0 alale. Kui vaatekauguse piiraja on sisse lülitatud (nagu see vaikimisi on), ei saa vaadet lohistada ega seda suurendada/vähendada. Vaatekauguse piiraja välja lülitamiseks vajuta ikooni, mis meenutab märki #.

- LMB  vajutamine kõverast väljaspool ja hiire lohistamine liigutab vaadet
- Suurendamiseks ja vähendamiseks kasuta + ja - ikoone.

Eritööriistad

Mutrivõtme ikooni vajutamisel ilmub menüü, kust saad vaate algseks tagasi muuta, määrata punktide interpolatsiooni ning kõverat lähtestada.

Sõlmegrupid

Kui su kaart kasvab, võib olla sõlmede organiseerimisel abi mõnede sõlmede grupeerimisest. Väikese nõuandena soovitame paigutada sõlmede 'ootmisliin' vasakult paremale, ülevalt alla. Allpool seletame ka, kuidas sõlmi grupeerida alamprogrammideks.

Sõlmede grupeerimine võib muuta sõlmede paigutust sõlmeredaktoris lihtsamaks ning muuta töö materjali või komposiitmise 'nuudliga' (sõlmede võrgustikuga) kergemaks. Sõlmede grupeerimine loob samuti sõlmegruppe (.blend-faili sees olevad NodeGroupid) või sõlmepuid (NodeTreed, mida kasutatakse lisamisel).

Kontseptuaalselt laseb "grupeerimine" määrata sõlmede *komplekti*, mida võid kasutada samamoodi, nagu see oleks "lihtsalt üks sõlm." Seejärel saad seda gruppi samas või mõnes teises .blend-failis uuesti kasutada.

Näide: kui oled loonud materjali, mida sooviksid kasutada teises .blend-failis, *võiksid* sa lihtsalt selle materjali ühest .blend-failist teise tõsta. Aga kui sa sooviksid luua uue materjali ning kasutada selle juures juba mõne valmis sõlmede võrgustiku haru? Sa võiksid selle haru taasluua. Või saaksid selle materjali uude .blend-faili lisada ning seejärel kleepida selle materjali sisse soovitud haru. Mõlemad võimalused töötavad, kuid ei ole erinevate .blend-failidega töötamisel just eriti efektiivsed. Mis siis, kui looksid "sügavusteravuse" komposiitmissõlmede võrgustiku ja sooviksid seda teises .blend-failis kasutada? Või kui sooviksid ühte ja sama tegevusteseeriat tosin korda rakendada? Ka sellisel juhul *võiksid* sa selle võrgustiku uuesti luua, kuid see poleks eriti efektiivne tegevus. Parem meetod nii materjali sõlmeharude kui komposiitmissõlmede võrgustike taaskasutamiseks oleks sõlmegruppide loomine.

Kui grupp määratakse, muutub see kindlaks objektiks; taaskasutatavaks tarkvarakomponendiks. Kui soovid, võid eirata implementatsiooni detaile ning kasutada seda (täpselt nii palju kordi kui soovid), milleks tarvis. Gruppe saab teistele failidele kasutatavaks muuta läbi [Blenderi teegi ning tavalise lisamise meetodi](#).

"PimedusseSulandumise" komposiitimisgrupp koosneb aja sõlmest (*Time*), mis annaks infot faktori (*Factor*) ja RGB-kõvera (RGB Curve) sõlmedele. RGB-kõvera sõlme kõver oleks nullis lame joon. Täpsemalt: iga sissetuleva X-i tulemuseks oleks nullväljund (must). Faktor algaks nulliga (sisendpilti ei muudeta) ja tõuseks üheni (täielik efekt ehk must). Sisendklipp sulanduks mustaks vastavalt aja sõlmes määratud kaadrite arvule.

Grupeerimine

Sõlmede grupi loomiseks vali sõlmeredaktoris sõlmed, mida soovid lisada, ja vajuta CtrlG või Space → node → make group.

Sõlmegrupi pealkirja riba on roheline. Kõik valitud sõlmed tehakse väikesteks ja pannakse grupi sisse.

Sõlmegruppidele antakse vaikimisi nimed *NodeGroup*, *NodeGroup.001* jne. Sõlmegrupis on nime väli, mille sisse vajutamisel saad grupi nime muuta. Nimeta sõlmegrupid ümber millekski tähenduslikuks. Sõlmegruppide lisamisel ühest .blend-failist teise ei erista Blender materjalisõlmede ja komposiitmissõlmede gruppe, seega soovitan võtta kasutusele mingi kindel nimetamissüsteem, mille alusel saaksid neid grupi tüüpe lihtsalt eristada. Näiteks materjalisõlmede harusid võid nimetada *Mat_XXX* ja komposiitmissõlmede võrke *Kom_XXX*.

Mida EI MAKSA gruppidesse lisada.

Pea meeles, et peamine idee seisneb selles, et iga grupp peaks põhimõtteliselt olema lihtsasti taaskasutatav, iseseisev komponent.

Materjalisõlmede grupid ei tohiks sisaldada materjali ja väljundi sõlmi.

Kui lisad gruppi materjali sõlme, siis ilmub materjalisõlm *kaks korda*: üks kord grupi sees ja teine kord grupist väljas uue materjali sõlmevõrgus.

Kui sa lisad grupi *sisse* väljundisõlme, siis ei tule grupist *välja* ühtegi väljundipesa!

Komposiitmissõlmede grupid ei saa sisaldada renderduskihi sõlme *Render Layer* (Blender ei lase sul seda sinna lisada) ning ei tohiks sisaldada väljundisõlme *Composite*. Ka siin tähendab lisatud väljundi sõlm (*Viewer*, *Split Viewer* jne) seda, et grupil ei ole enam väljundipesa.

Sõlmegruppide muutmine

Kui sõlmegrupp on valitud, muudab Tabi vajutamine sõlme aknaavaks ning näitatakse selles sisalduvaid üksikuid sõlmi. Neid saab ringi liigutada, üksikuid kontrollereid muuta, nende vahelisi siseseid lõimi ümber tõsta jne. Täpselt nii, nagu saaksid teha siis, kui nad oleksid sinu tavapärasel redaktoriaknas. Sa ei saa neid ühendada lõimega mõne grupist välja jääva sõlme külge, selleks pead sa kasutama grupisõlme külgedel olevaid väliseid pesi. Sõlmede gruppi lisamiseks või sealt eemaldamiseks pead grupi lõhkuma.

Sõlmegrupi lõhkumine

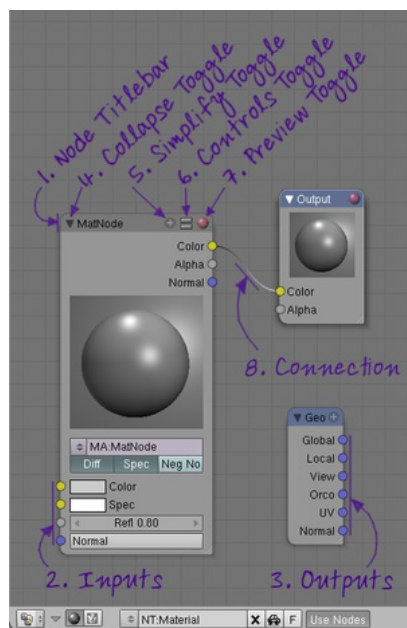
Kiirvalik AltG hävitab grupi ja asetab üksikud sõlmed tagasi redaktorisse. Kõik sisemised ühendused jäävad alles ning nüüd saad sa ka sisemisi sõlmi teiste redaktoris olevate sõlmedega ühendada.

Sõlmegruppide lisamine

Kui oled mõne sõlmepuu (*NodeTree*) oma .blend-faili lisanud, saad seda sõlmeredaktoris kasutada, vajutades Space → Add → Groups ning valides seejärel soovitud grupi. Grupi "juhtpaneeliks" on iga grupi sees oleva sõlme enda kontrollid. Neid saad grupiga töötamisel muuta nagu iga teise sõlme omasidki.

Sõlme juhtelemendid

See lehekülg selgitab sõlme juhtelemente



Sõlme peamised juhtelemendid

Nimeriba

Selles on sõlme nimi koos erinevate sõlme kokku voltimise nuppudega.

Sisendi pesad

Sõlme vasakul küljel on sisendi pesad:

- *sinised pesad* võtavad vastu vektoreid
- *kollased pesad* võtavad vastu värve
- *hallid pesad* võtavad vastu üksikväärtusi (nagu alfa)

Väljundi pesad

Sõlme paremal küljel on väljundi pesad:

- *sinised pesad* toodavad vektoreid
- *kollased pesad* toodavad värve
- *hallid pesad* toodavad üksikväärtusi (nagu alfa)

Pildi eelvaade / kõver

Sõlme sees on ala, mis näitab sõlmest väljuva pildi eelvaadet või kõveraid, mille abil saab määrata sõlme käitumist (näiteks RGB-sõlmes).

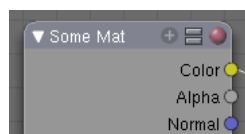
Nupud ja menüüd

Pildi eelvaate all on nupud ja menüüd, mis määravad sõlme käitumist.

Lõimed

Kõver joon kujutab endast väljundi pesa ja sisendi pesa vahelist ühendust. Pesade tüübid peavad olema vastavuses. Aktiivse sõlmega seotud ühendused on parema nägemise nimel esile tõstetud.

Kokku voltimise nupud



Sõlme päis.

Sõlme päises on kuni neli nähtavat juhtelementi (vt pilt *Sõlme päis*). Nendele vajutamine mõjutab seda, palju infot sõlm kuvab.

Sõlme nupp (▼►)

Vasakul olev nool voldib sõlme täielikult kokku või lahti.

Pesa nupp (+)

Paremal olev plussmärgiga nupp näitab/peidab kasutamata sisendi/väljundi pesasid.

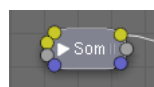
Menüü nupp (≡)

Kahe joonega nupp keskel näitab/peidab kõiki sõlmes olevaid juhtelemente.

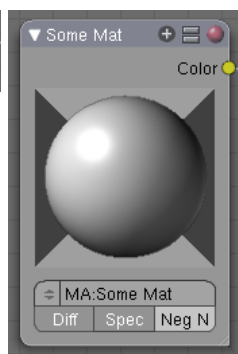
Eelvaate pildi nupp ()

Nimeriba kõige paremas servas olev keraga nupp peidab/näitab eelvaate pilti. The sphere button on the far right of the titlebar hides/unhides preview image. Kui kera on **punane** võib sel olla kolm põhjust:

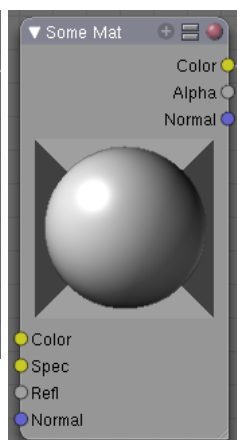
- see on ainus sõlmedaktori töötav väljundi sõlm.
- see on [materjali sisendi sõlm \(Material\)](#), millele on määratud materjal (MA:).



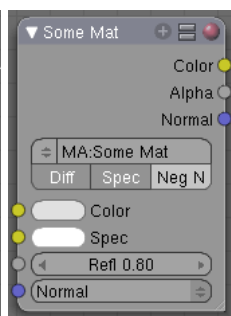
Kokku voltimise nool.



Plussmärk.



Menüü nupp.

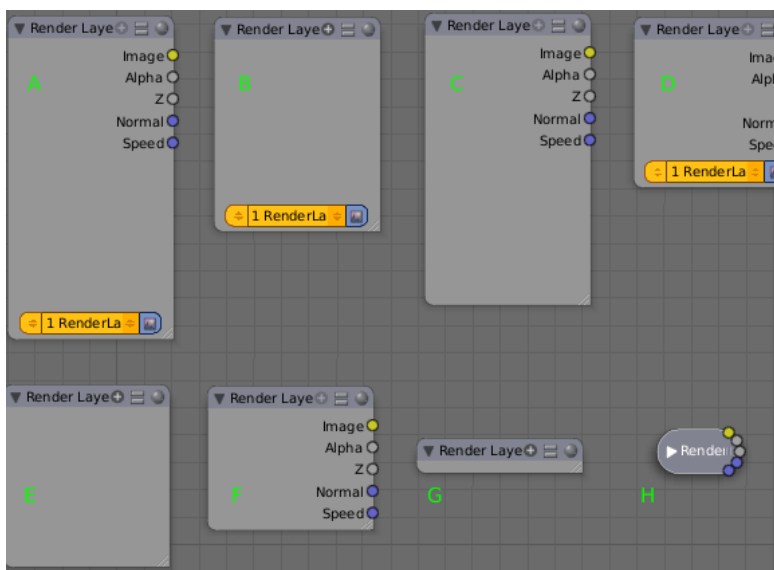


Kera.



Koos.


Viimast kolme saab omavahel koos kasutada. Noolt, mis voldib kokku terve sõlme, saab kasutada ainult koos plussmärgiga (vt pilt *Koos*).



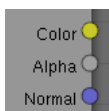
Sõlme ülemised suuruse nupud

- A) Tavaline, B) + märki on vajutatud, C) = märki on vajutatud, D) Kera on vajutatud, E) + ja = märke on vajutatud, F) = ja kera on vajutatud, G) Kõiki kolme on vajutatud H) Noolt on vajutatud.

Sõlme suuruse muutmine

Täpsemalt saab sõlme suurust mingil määral muuta, vajutades selle alumisel paremal nurgal (kus on väikesed kaldjooned) klahvi LMB  ja lohistades hiirt.

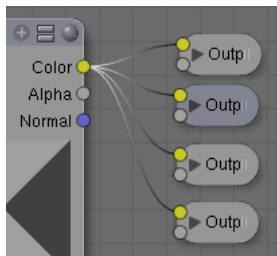
Pesad



Sõlme pesad.

Igal sõlmede aknas asuval sõlmel on "pesad" (sageli nimetatakse neid ka "ühendusteks" või sisendiks-väljundiks), mis kujutavad endast väikeseid värvilisi ringe, millesse ühendatakse sisend- ja väljundandmed (vt pilt *Sõlme pesad*).

Sõlme vasakul küljel olevad pesad tähistavad *sisendeid*,"*paremale jäävad pesad väljundeid*."



Sõlmede ühendamine.

Elu lihtsustamiseks on pesad värviga tähistatud vastavalt sellele, mis tüüpi informatsiooni nad eelistavad saata või vastu võtta. Värv on kolm:

Kollased pesad

Tähistab seda, et pesa võetakse vastu või sellest saadetakse sõlme **värvi** informatsiooni.

Hallid pesad

Tähistab väärtustega (**numbrilist**) informatsiooni. See võib olla kas üksik numbriline väärtus või "väärtuste kaart". (Väärtuste kaarti võid ette kujutada halltoonides kaardina, kus erinev heleduse/tumeduse hulk väljendab iga eri punkti väärtust.) Kui "väärtuste kaardi" pesas kasutada üksikut väärtust, kasutavad kõik kaardi punktid sama väärtust. Tavaline kasutamisiis: alfakaardid ja sõlmede numbrilise väärtusega väljad.

Sinised/lillad pesad

Tähistab **vektor-/koordinaat-/normaalide** informatsiooni.

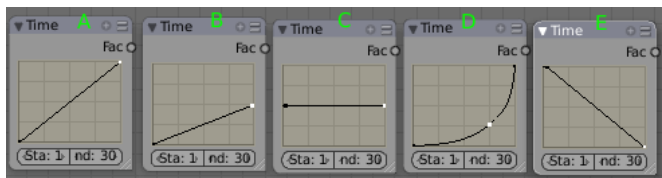
Kui sa ei kasuta *konverterit*, millest räägime hiljem, tuleb sõlmede vahel kollased pesad ühendada kollastega, hallid hallide ja sinised sinistega.

Värvi kõrval näed sõlmes vastava pesa nime. Kuigi see ei ole alati nii, võib nime pidada vihjeks selle kohta, mis laadi infot pesa eeldab. Kuid see ei tähenda, et info *peab* seda tüüpi olema. Ma võin näiteks ühendada alfata sisaldava pesa nimega (Alpha) halli värvi pesa mõne materjalisõlme halli peegelduvuse (Reflection) pesa ning saavutada ikkagi tulemust. Kõige olulisem on selle puhul, et tegemist on "hall-hall" ühendusega.

On erandeid, mille puhul saad ilma konverteriteta segada kollast (*nt* värviline pilt) ja halli (*nt* halltoonid). Tavaliselt lisab Blender ise konverteri, kui seda vaja on, seega eksperimenteerige vabalt. Saad kasutada vaataja väljundisõlmi "Viewer", mida seletame hiljem, nägemaks, kas ja kuidas su eksperiment töötab.

Kõverad

Mõnedel sõlmedel on kõveratega piirkonnad, mis tõlgivad sisendi väärtuse väljundi väärtuseks. Seda kõverat saab muuta, vajutades mõnele kontrollpunktile ja seda liigutades või kontrollpunkte lisades. Mõned näited on toodud allpool:



Sõlmekõvera muutmine.

Iga kõver on algses sirge joon tõusuga 1. Sellisel kõveral on mõlemas otsas kaks väikest musta värvi juhtpunkti. Kui vajutada juhtpunkti peal LMB , siis see tähistatakse ja muutub valgeks.

Kõvera muutmine määrab, kuidas väljundit genereeritakse. Sisend, X, liigub tavaliselt lineaarselt (korrapäraste vahemaadega) mööda **alumist** telge. Liigu üles kõverani ning sealt edasi **paremale**: see määrab vastava X-i väljundi Y. Seega teise näite puhul, kus X liigub mööda alumist serva punktist 0 punktini 1.0, varieerub Y-i väärtus 0.0 ja 0.5 vahel. Kolmandas, kus X liigub mööda alumist serva punktist 0.0 punktini 1.0, jääb Y-i väärtuseks pidevalt 0.5. Seega on ülemisel pildil olevatel kõveratel mõju ajale järgmine: **A** ei mõjuta, **B** aeglustab, **C** peatab, **D** kiirendab ja **E** pöörab aja ringi.

"Kõverate" vidin on Blenderi kasutajaliidesesse sisse ehitatud ning seda saab igal pool kasutada tingimusel, et vidinale antakse vajalikud kõvera andmed. Praegu kasutatakse seda sõlmeredaktoris ja UV aknas.

See vidin kaardistab sisendi väärtuse horisontaalselt ning tekitab kõvera kõrgusest sõltuva uue väärtuse.

Märkus: see, et igal pildil on üks kõvera punkt "valge", *ei oma* tähtsust: see tähistab lihtsalt seda punkti, mille seda õpetust ettevalmistanud autor viimati valis. Antud juhul on oluline *kõvera* kuju, mitte selle määramiseks kasutatud kontrollpunktide asukoht (või nende värv).




RGB-kõverad

Ühes kõveravidinas saab muuta mitut erinevat kõverat. Selle tavaliseks näiteks on RGB-kõver, millel esimeseks kõveraks on "kombineeritud" tulemus või "värv" ("C") ning milles on olemas ka eraldi kõverad iga RGB komponendi (punane-roheline-sinine) muutmiseks. Kõik neli kõverat on korraga aktiivsed, kuid esimesena kasutatakse arvutamiseks "C" kõverat.

Kõverapunktide valimine


- LMB valib alati ühe punkti ja tähistab ülejäänute valiku.
- Hoia vajutamise ajal all nuppu Shift, et valitud punktide arvu suurendada või vähendada.

Kõverate muutmine

- LMB  vajutamine ja punkti lohistamine liigutab punkte.
- Kõvera peal LMB  vajutamine lisab uue punkti.
- Ühe punkti täpselt teise peale lohistamine liidab nad üheks punktiks.
- Lohistamise ajal ⇧ Shift all hoidmisel nakkuvad punktid ruudustikuga.
- Ctrl LMB  lisab punkti.
- Valitud punktide kustutamiseks vajuta ikooni X.

Vaate muutmine

Vaikimisi vaade keskendub 0.0–1.0 alale. Kui vaatekauguse piiraja on sisse lülitatud (nagu see vaikimisi on), ei saa vaadet lohistada ega seda suurendada/vähendada. Vaatekauguse piiraja välja lülitamiseks vajuta ikooni, mis meenutab märki #.

- LMB  vajutamine kõverast väljaspool ja hiire lohistamine liigutab vaadet
- Suurendamiseks ja vähendamiseks kasuta + ja - ikoone.

Eritööriistad

Mutrivõtme ikooni vajutamisel ilmub menüü, kust saad vaate algseks tagasi muuta, määrata punktide interpolatsiooni ning kõverat lähtestada.

Sõlmede kasutamine

Sõlmede lisamine


Sõlmeredaktorisse lisatakse sõlmi kahel moel:

- Kasutades sõlmeredaktori tööriistaribal olevat lisamise menüüd Add ja valides sealt soovitud sõlmetüübi või
- Hoides kursorit sõlmeredaktori aknas, vajutades kiirvalikut ⇧ ShiftA ning valides hüpikmenüüst Add mõne sõlme.

Sõlmede paigutamine

Üldiselt proovi paigutada sõlmi aknas nii, et pildi arvutamine toimuks vasakult paremale, ülalt alla. Sõlme liigutamiseks vajuta mõnele selle neutraalsele alale ja lohista siis ringi. Sõlme peale võib vajutada peaaegu ükskõik kuhu ning seejärel seda ringi lohistada; ühendused proovivad ennast võimalikult hästi Bezier' kõveratena ümber kohendada.



Sõlmede ühendamine

Vajuta soovitud pesale klahviga LMB  ja lohista hiirt: näed, kuidas pesast tuleb välja haru. Seda kutsutakse "lõimeks".

Jätka lohistamist ja ühenda see lõime mõne teise sõlme sisendi pesaga, lastes seejärel klahvi LMB  lahti.

Sellisel juhul saadetakse iga väljundpesa edasi mööda sõlmi. Kuid igasse sisendpesasse saab ühenduda ainult üks lõim.

Sõlmede lahtiühendamine

Pesadevahelise ühenduse katkestamiseks vajuta lõigatava lõime lähedale tühjale alale klahviga LMB  ja lohista hiirt - hiire kursori kõrvale ilmub pisike noatera ikoon. Liiguta hiir üle lõigatava lõime ning lase siis klahv LMB  lahti.

Sõlmede duplitseerimine

Vajuta soovitud sõlmele, seejärel kasuta kiirvalikut ⇧ ShiftD ning liiguta hiirt, et hiirekursori alla ilmuks valitud sõlme koopia.

Ahhaa!

Kui sa mõnd sõlme duplitseerid, siis asetatakse uus sõlm *täpselt* vana peale. Kui sõlme sinna jätadki (ja seda võib julgelt teha), siis sa lihtsalt ei näe, et seal on tegelikult *kaks* sõlme! Jah, sõlme vari võib olla natuke tumedam kui enne, kuid enam ei ole see visuaalselt selge, mis on millega ühendatud. Kui tekib kahtlusi, haara sõlmest kinni ja liiguta seda natuke, et näha, kas selle all miski ennast varjab.

Sõlmegrupid

Sõlmede grupeerimine võib muuta sõlmede paigutust sõlmeredaktoris lihtsamaks ning muuta töö materjali või komposiitmise 'nuudliga' (sõlmede võrgustikuga) kergemaks. Sõlmede grupeerimine loob samuti sõlmegruppe (.blend-faili sees olevad NodeGroupid) või sõlmepuid (NodeTreed, mida kasutatakse lisamisel).

Kontseptuaalselt laseb "grupeerimine" määrata sõlmede *komplekti*, mida võid kasutada samamoodi, nagu see oleks "lihtsalt üks sõlm." Seejärel saad seda gruppi samas või mõnes teises .blend-failis uuesti kasutada.

Näide: kui oled loonud materjali, mida sooviksid kasutada teises .blend-failis, *võiksid* sa lihtsalt selle materjali ühest .blend-failist teise tõsta. Aga kui sa sooviksid luua uue materjali ning kasutada selle juures juba mõne valmis sõlmede võrgustiku haru? Sa võiksid selle haru taasluua. Või saaksid selle materjali uude .blend-faili lisada ning seejärel kleepida selle materjali sisse soovitud haru. Mõlemad võimalused töötavad, kuid ei ole erinevate .blend-failidega töötamisel just eriti efektiivsed. Mis siis, kui looksid "sügavusteravuse" komposiitmissõlmede võrgustiku ja sooviksid seda teises .blend-failis kasutada? Või kui sooviksid ühte ja sama tegevusteseeriat tosin korda rakendada? Ka sellisel juhul *võiksid* sa selle võrgustiku uuesti luua, kuid see poleks eriti efektiivne tegevus. Paremad meetod nii materjali sõlmeharude kui komposiitmissõlmede võrgustike taaskasutamiseks oleks sõlmegruppide loomine.

Kui grupp määratakse, muutub see kindlaks objektiks; taaskasutatavaks tarkvarakomponendiks. Kui soovid, võid eirata implementatsiooni detaile ning kasutada seda (täpselt nii palju kordi kui soovid), milleks tarvis. Gruppe saab teistele failidele kasutatavaks muuta läbi [Blenderi teegi ning tavalise lisamise meetodi](#).

Grupeerimine

Panel: Sõlmeredaktor

Menu: ⇧ ShiftA → Groups (grupid)

Sõlmede grupi loomiseks vali sõlmeredaktoris sõlmed, mida soovid lisada, ja vajuta CtrlG või Space → *node* → *make group*. Sõlmegrupi pealkirja riba on roheline. Kõik valitud sõlmed tehakse väikesteks ja pannakse grupi sisse. Sõlmegruppidele antakse vaikimisi nimed *NodeGroup*, *NodeGroup.001* jne. Sõlmegrupis on nime väli, mille sisse vajutamisel saad grupi nime muuta. Nimeta sõlmegrupid ümber millekski tähenduslikuks. Sõlmegruppide lisamisel ühest .blend-failist teise ei erista Blender materjalisõlmede ja komposiitmissõlmede grupe, seega soovitan võtta kasutusele mingi kindel nimetamissüsteem, mille alusel saaksid neid grupi tüüpe lihtsalt eristada. Näiteks materjalisõlmede harusid võid nimetada *Mat_XXX* ja komposiitmissõlmede võrke *Kom_XXX*.



Mida EI MAKSA gruppidesse lisada (kõigi sõlmeredaktori tüüpide puhul)

Pea meeles, et peamine idee seisneb selles, et iga grupp peaks põhimõtteliselt olema lihtsasti taaskasutatav, iseseisev komponent. Sõlmede grupid **ei tohiks sisaldada**:

Allikasõlmi

Kui lisad grupi allika sõlme, siis ilmub allika sõlm *kaks korda*: üks kord grupi sees ja teine kord grupist väljas uues sõlmevõrgus.

Allika sõlmed on näiteks: *Materjali sõlm* (materjalide sõlmeredaktoris) ja *Rendersukihtide sõlm* (komposiitimisredaktoris).

Väljundi sõlme

Kui sa lisad grupi *sisse* väljundisõlme, siis ei tule grupist *välja* ühtegi väljundipesa!

Väljundi sõlmed on näiteks: väljundi sõlm *Output* (materjalide sõlmeredaktoris) ja vaataja sõlm *Viewer* (komposiitimisredaktoris).

Sõlmegruppide muutmine

Kui sõlmegrupp on valitud, muudab ⇆ Tab vajutamine sõlme aknaavaks ning näitatakse selles sisalduvaid üksikuid sõlmi. Neid saab ringi liigutada, üksikuid kontrollereid muuta, nende vahelisi siseseid lõimi ümber tõsta jne. Täpselt nii, nagu saaksid teha siis, kui nad oleksid sinu tavapärase redaktoriaknas. Sa ei saa neid ühendada lõimega mõne grupist välja jääva sõlme külge, selleks pead sa kasutama grupisõlme külgedel olevaid väliseid pesi. Sõlmede gruppi lisamiseks või sealt eemaldamiseks pead grupi lõhkuma.

Sõlmegrupi lõhkumine

Kiirvalik AltG hävitab grupi ja asetab üksikud sõlmed tagasi redaktoris. Kõik sisemised ühendused jäävad alles ning nüüd saad sa ka sisemisi sõlmi teiste redaktoris olevate sõlmedega ühendada.

Sõlmegruppide lisamine

Kui oled mõne sõlmepuu (*NodeTree*) oma .blend-faili lisanud, saad seda sõlmeredaktoris kasutada, vajutades Space → *Add* → *Groups* ning valides seejärel soovitud grupi. Grupi "juhtpaneeliks" on iga grupi sees oleva sõlme enda kontrollid. Neid saad grupiga töötamisel muuta nagu iga teise sõlme omasidki.

Materjalisõlmede tüübid

See peatükk on organiseeritud sõlme tüübi järgi, mis on grupeeritud sarnaste funktsioonide järgi:

- [Sisend \(Input\)](#) - Toob sõlmekaardile mõne materjali või komponendi.
- [Väljund \(Output\)](#) - Näitab töösolevat tulemust väikese pildina.
- [Värv \(Color\)](#) - Manipuleerib materjali värve.
- [Tipp \(Vector\)](#) - Muudab seda, kuidas valgus materjalilt peegeldub.
- [Konverterid \(Convertors\)](#) - Teisendab värve teisteks materjali värvideks.
- [Grupid \(Groups\)](#) - Kasutaja poolt määratud sõlmegrupid.
- [Dünaamilised \(Dynamic\)](#) - Pythoni skriptidega määratud sõlmed. Neid sõlmi tuntakse ka nimetusega PyNodes.

Materjali sisendi sõlm (*Input*)

Alustamiseks vajalik materjal luuakse materjalide paneelis. Seal lülitatakse sisse sõlmede nupp Nodes, mis lisab materjali sõlmeredaktori akna päises olevasse sõlmitavate materjalide nimekirja. Sõlmekaardi võimalikeks sisenditeks on:

- Väärtus (*Value*)
- Värv (*Color*)
- Tekstuur (*Texture*)
- Geomeetria (*Geometry*)
- Materjal (*Material*)
- Kaamera andmed (*Camera Data*)

Materjalisõlm (*Material*)

Panel: Sõlmeredaktor → Materjalisõlmed

Menu: ⇧ ShiftA → *Input* → Material



Materjalisõlm
Material

Materjalisõlme kasutatakse sõlmede võrku materjali lisamiseks. Materjalid võivad olla kõike alates lihtsast varjutusest kuni mitmekihiliste tekstuuridega materjalideni. See sõlm avaldab materjali peamised omadused (värv, alfa, tipu normaal) sõlmekaardile.



Väljund

Materjalid võivad väljastada värvi (mille hulka käib ka varjutaja ja kõik sellele määratud tekstuurid), alfat ja tekstuuri põhjal arvutatud lõpliku normaali.

- Värv (*Color*)
- Alfa (*Alpha*)
- Normaal (*Normal*)

Sisend

Materjalid võivad sisse võtta värve, hajuvat värvi, läike värvi, peegeldavuse väärtust ja normaali.

- Värv (*Color*) - alusvärv. Seda saab määrata
 - käsitsi, vajutades pesa kõrval olevat värvinäidisel LMB , valides avanevast juhtpaneelist värvi ja vajutades ↵ Enter
 - materjali paneeliit, muutes aktiivse materjali omadusi
 - sisendisse lisatud RGB-värvi generaatorist.
- Läige (*Spec*) - värv, mida materjal peegeldab, kui valgusallikas on pinnaga sobiva nurga all. Seda saab määrata
 - mõnest teisest sõlmest sisendisse tuleva väärtusega
 - vajutades värvinäidise peal LMB , ja valides selle värvi paneelist.
- Peegeldavus (*Refl.*) - nurk, milleni materjal peegeldab valgust ja annab edasi selle värvi. See väärtus võib tulla mõnest teisest sõlmest või olla määratud käsitsi.
- Normaal (*Normal*) - valgustustingimused.

Juhikud

Materjali väli (MA:Material)

Selle kaudu saad otsida ja valida materjale.

Hajumise märkeruut (*Diff*)

Lülita hajususe värv sisse ja välja.

Läike märkeruut (*Spec*)

Lülita läike arvutamine sisse ja välja.

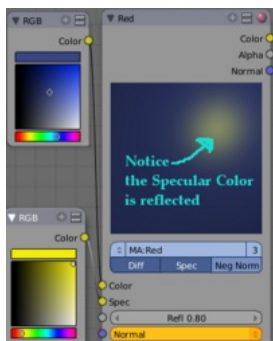
Negatiivi lüliti (*Neg*)

Kui see on sisse lülitatud, pööratakse materjali sisse tulev normaal tagurpidi (mis loomulikult tähistab kombinatsiooni 3D-normaalist, mille materjalile annab 3D-objekt, ning normaali sisendist).

Normaali tühistamine

Normaali sisendi pesa ei sega mitte mingil kujul sissetulevat väärtust päris geomeetriaga. Sellesse pesasse sisestatud geomeetria (*Geometry*) tühistab normaali valgustustingimused.

Materjalisõlme kasutamine läikega



Läiget kasutav materjalisõlm

Et materjalisõlm tekitaks värvi, pead määrama vähemalt baas-sisendvärvi ja soovitatavalt ka läike värvi. Läike värv on see värv, mis intensiivse valguse all särab.

Vaata näiteks paremal olevat pisikest sõlmekaarti. Baasvärv (tumesinine) on materjalisõlme sisestatud RGB värvi genereerivast sõlmest, kasutades värvipesa Color. Läike värv (kollane) on sisestatud läbi läike sisendi Spec. Tavalise valguse tingimustes lamedal pinnal tekitab see materjal sügavsinise värvi, millesse hakkab sisse sulanduma kollast läikevärvust, mida lähemale sa valgusallika peegelduspunktile jõuad.

Läike sisse lülitamine

Läike nägemiseks pead selle sisse lülitama sinise läike nupuga Spec, mis on sõlmes täpselt värviparameetri all.

Väärtuse sõlm (Value)



Väärtuse sõlmel ei ole sisendeid; see lihtsalt väljastab numbrilise väärtuse (ujukomaarvu 0.00-st kuni 1.00-ni), mis on sisestatud tema numbriväljale.

Kasuta seda sõlme teiste sõlmede väärtuse või faktori sisendpesadesse püsivalt määratud väärtuse andmiseks.

RGB sõlm (RGB)

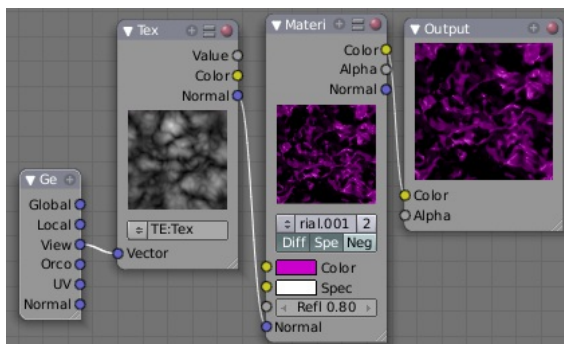


RGB sõlmel ei ole sisendeid. See väljastab parajasti valitud värvi, nagu ülalolevas raamis näha. Paremal olevas näites on valitud kerge punase tooniga hall.

Värvi heleduse või küllastatuse muutmiseks vajuta LMB ruudukujulise värvisulatuses. Hetke küllastatust näitab sulatuses olev väike ring. Värvi tooni muutmiseks vajuta vikerkaarevärvides ülemineku peale.

Tekstuurisõlm (Texture)

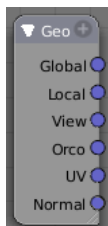
Sõlm sisaldab antud .blend-failis olevat tekstuuride nimekirja, mida pärast valimist saab sõlmest väärtuse (*value*) ja/või värvi (*color*) pesa abil edasi saata.



Näites on paremas servas lõplik materjal, nagu vaataja seda näeb – see koosneb tavalisele lillale materjalile lisatud pilve tekstuurist ning annab sametja efekti.

Pane tähele, et sa võid kasutada mitut tekstuuri sisendsõlme. (Vanas) paneelide süsteemis määrati mitmele tekstuurile eraldi kanalid ning igat kanalit tuli materjalile lisamiseks sisse või välja lülitada. Sõlmedega lisad lihtsalt tekstuuri kaardile ning ühendad selle teiste sõlmedega.

Geomeetria sõlm (*Geometry*)



Geomeetria sõlm
(Geometry)

Geomeetria sõlme kasutatakse pinnalt valguse peegeldamise määramiseks. Seda sõlme kasutatakse selleks, et muuta materjalide normaalide käitumist vastavalt valguse tingimustele.

Kasuta seda sõlme, et saada materjalisõlme normaalivektorite andmed, et näha, kuidas materjal erinevate valgustingimuste all välja näeb (nt särab või peegeldab valgust). Valikuteks on:

Global (globaalne)

Pinna globaalne asukoht.

Local (kohalik)

Pinna kohalik asukoht.

View (vaade)

Pinna vaatamise nurk.

Orco (originaalkoordinaadid)

Kasuta võre originaalkoordinaate.

UV

Kasuta võre UV-koordinaate.

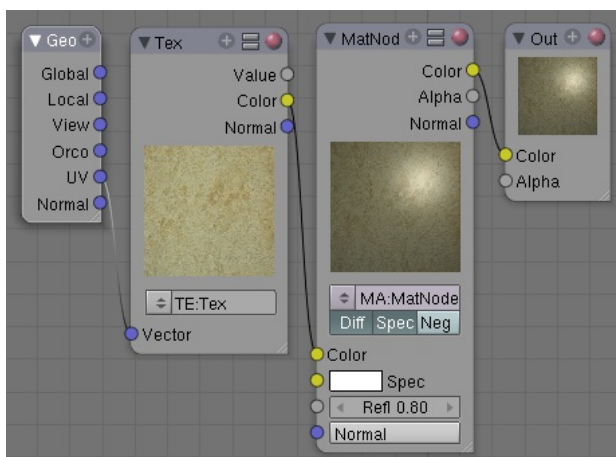
Normal (normaal)

Pinnanormaalid; tasapinnaline, ülevalt ja paremalt tulevad valgused, mis peegelduvad pinnalt.

Note

Need on täpselt samad sätted mis [tekstuuri](#) puhul [laotuse](#) paneelil, kuigi mõned neist on puudu (näiteks pinge (Stress) ja puutuja (Tangent)). Tavaliselt kasutaksid sa seda sõlme [tekstuurisõlme](#) sisendina.

UV-pilti kasutava geomeetria sõlme näide



Sõlmekaart UV-laotusega pilttekstuuri renderdamiseks.

Näiteks: UV-laotusega pildi renderdamiseks kasutaksid sa UV väljundit ja ühendaksid selle mõne tekstuurisõlme vektorsisendiga (Vector). Seejärel ühendaksid tekstuuri sõlme värvi väljundi materjali sõlme värvi sisendisse, mis vastaks paneeli laotuse sättele Map To.

Lisaomadustega materjal (*Extended Material*)

Lisab materjalisõlmele rohkem sisend- ja väljundkanaleid.

Sisend

Värv (Color)

Selle juures on värvinäidis, mis laseb sul otse sõlme sees värvi valida.

Peegeldusvärv (Mirror Color)

Keskkonnad (Ambient)

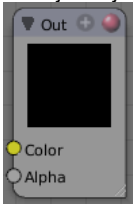
Kiirgus (Emit)
Läike läbipaistvus (SpecTra)
Kiirtepeegeldus (Ray Mirror)
Alfa (Alpha)
Läbikumavus (Translucency)

Väljund

Hajusus (Diffuse)
Läige (Spec)
Kaudvari (AO)

Kaamera andmed (*Camera Data*)

View Vector (vaate vektor)
View Z Depth (vaate sügavus)
View Distance (vaate kaugus)

Materjali väljundsõlm (*Output*)Väljundsõlm
(Output)

Suvalisel hetkel võib sul tekkida soov näha käsil oleva töö tulemust. Eriti pärast seda, kui mõni sõlm on oma töö teinud. Selleks pead vedama soovitud sõlme väljundi pesast uue lõime mõne väljundisõlme (*Output*) sisendi pessa ning näedki tillukest pilt.

Ühenda ka alfakanal, kui läbipaistvust näha soovid.

Lõplik väljund

Sellel ainsal väljundisõlmel, mida materjal lõpuks kasutab (st see ei ole mõeldud eelvaateks), on üleval paremal nurgas **punane kera**.

Materjali värvisõlmed (*Color*)

Mix - sega

See sõlm segab alusvärvi või pildi (mis on lõimitud ülemisse pesasse) kokku teise värvi või pildiga (alumises pesas), töödeldes mõlema pildi üksikud vastavuses olevad pikslid omavahel läbi. Seda, kuidas tulemuseks olev pilt valmib, saab valida rippmenüüst. Segamissõlme tulemuseks oleva pildi suurus (väljundi resolutsioon) sõltub baaspildi suurusest. Segatakse ka alfa ja Z-kanalid (komposiitmissõlmedel).

Blenderiga tuleb kaasa mitte üks ega kaks, vaid (loe üle, kui ei usu) kuusteist segamise võimalust:

Sega (<i>Mix</i>)	Taustapiksel kaetakse esiplaani omaga alfa väärtuseid kasutades.
Liida (<i>Add</i>)	Pikslid liidetakse kokku. Teguri väli Fac määrab, palju teist pesa liidetakse. Tulemus on ere. Selle režiimi "vastand" on lahutamisrežiim <i>Subtract</i> .
Lahuta (<i>Subtract</i>)	Esiplaani piksel (alumine pesa) lahutatakse taustapikslist. Tulemus on tume. Selle režiimi "vastand" on liitmissežiim <i>Add</i> .
Korruta (<i>Multiply</i>)	Tulemuseks on enamasti tumedam piksel kui algsed pikslid (välja arvatud siis, kui üks neist on valge=1.0). Täiesti valged kihid ei mõjuta tausta üldse. Täiesti mustade kihtide tulemuseks on must. Selle režiimi "vastand" on varjutamisrežiim <i>Screen</i> .
Ekraan (<i>Screen</i>)	Mõlema piksli väärtus pööratakse ümber, korrutatakse omavahel ning tulemus pööratakse uuesti ümber. Tulemuseks on enamasti heledam piksel kui algsed pikslid (välja arvatud siis, kui üks nendest on 0.0). Täiesti mustad kihid ei muuda tausta üldse (ja vastupidi), täiesti valgete kihtide tulemuseks on valge. Selle režiimi "vastand" on korrutamissežiim <i>Multiply</i> .
Kihinda (<i>Overlay</i>)	Kombinatsioon varjutamisrežiimist <i>Screen</i> ja korrutamissežiimist <i>Multiply</i> , sõltub alusvärvist.
Jaga (<i>Divide</i>)	Taustapiksel (ülemine pesa) jagatakse teisega: kui see on valge (= 1.0), siis esimest ei muudeta; mida tumedam on teine, seda heledam on tulemus (jagamine 0.5-ga (keskmine hall) on sama, mis korrutamine 2.0-ga); kui teine on must (= 0.0, nulliga jagamine on võimatu!), siis Blender ei muuda taustapikslit.
Erista (<i>Difference</i>)	Mõlemad pikslid lahutatakse teineteisest ning võetakse nende absoluutväärtus. Seega näitab tulemus mõlema parameetri vahelist kaugust: must tähistab võrdseid värve, valge vastandvärve (üks on must, teine valge). Tulemus on paljudel juhtudel väga imelik. Seda režiimi saab kasutada aluspildi osade värvi pühkimiseks ning kahe pildi võrdlemiseks (tulemus on must, kui nad on samad).
Tumenda (<i>Darken</i>)	Mõlemat pikslit võrreldakse teineteisega, võetakse väiksem väärtus. Täiesti valged kihid ei mõjuta tausta üldse ning täiesti mustade kihtide tulemus on must.
Helenda (<i>Lighten</i>)	Mõlemat parameetrit võrreldakse teineteisega, võetakse suurem väärtus. Täiesti mustad kihid ei mõjuta pilti üldse ja täiesti valgete kihtide tulemus on valge.
Helesta (<i>Dodge</i>)	Teatav tagurpidi pööratud korrutamissežiim <i>Multiply</i> (korrutamine on asendatud "vastandväärtuse" jagamisega). Tulemuseks on pildil olevad heledamad alad.
Põleta (<i>Burn</i>)	Teatav tagurpidi pööratud varjutamisrežiim <i>Screen</i> (korrutamine on asendatud "vastandväärtuse" jagamisega). Tulemuseks on tumedam pilt, sest pilt põletatakse paberil, eee... pildile (kus ma olen alles vana).
Värv (<i>Color</i>)	Lisab pikslile värvi ning toonib üldpildi selle värviga. Kasuta seda režiimi pildi tonaalsuse nihutamiseks.
Heledus (<i>Value</i>)	Mõlema piksli RGB väärtused konverteeritakse HSV väärtusteks. Mõlema piksli heledus segatakse, taustapildi toon ja küllastus kombineeritakse segatud heledusega ja konverteeritakse tagasi RGB-sse.
Värviküllastus (<i>Saturation</i>)	Mõlema piksli RGB väärtused konverteeritakse HSV väärtusteks. Mõlema piksli värviküllastus segatakse, taustapildi toon ja heledus kombineeritakse segatud küllastusega ja teisendatakse tagasi RGB-sse.
Toon (<i>Hue</i>)	Mõlema piksli RGB väärtused konverteeritakse HSV väärtusteks. Mõlema piksli toon segatakse, taustapildi heledus ja küllastus kombineeritakse segatud tooniga ja teisendatakse tagasi RGB-sse.

Värvikanalid

On kaks moodust, kuidas väljendada kombineeritud kanaleid, mille tulemuseks on värv: RGB või HSV. RGB tähistab punast-rohelist-sinist piksliformaati ja HSV toon-küllastus-heledus piksliformaati.

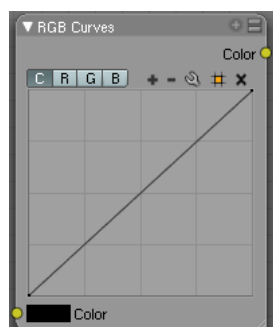
A

Vajuta rohelist alfa nuppu, kui tahad sundida segamissõlme kasutama teise (alumise) sõlme alfa (läbipaistvuse) väärtusi. Kui see on sisse lülitatud, vastab tulemuseks saadava pildi alfakanal mõlema pildi vastavatele kanalitele. Muidu (kui see pole sisse lülitatud ja tuli on roheline) on saadava pildi alfakanal sama mis aluspildil (ülemine sisendi pesa).

Fac

Alumise pesa sissesegamise hulk sõltub faktori väljast (Fac:). Väärtus 0 ei kasuta alumist pesa üldse ning väärtus 1.0 kasutab seda täielikult. Segamisrežiimis Mix tähendab 50:50 (0.50) kahe sisendi võrdset segamist, kuid liitmissežiimis *Add* tähendab 0.50 seda, et lisatakse ainult pool teise pesa mõjust.

RGB kõverad (*RGB Curves*)

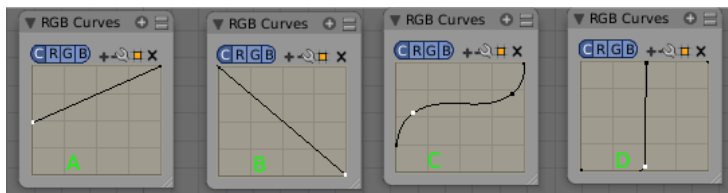


RGB kõverate sõlm RGB Curves

Selle sõlme abil saad määrata iga värvi komponentkanali (RGB) või nende komposiidi (C) kohta Bezier' kõvera, mis mõjutab sisendit (tähistatud all mööda x-telge) ja tekitab väljundväärtuse (y-telg). Vaikimisi on see ühtlaselt tõusev sirgjoon, mistõttu tähendab x-telje 0.5 y-telje 0.5 väljundit. Kui klõpsata hiirega kõveral ja seda lohistada, luuakse uus kontrollpunkt ja kõvera kuju muutub. Kui tahad punkti kustutada, vajuta nuppu X.

Nupud C R G B näitavad vastava kanali kõverat. Tehes näiteks komposiidi (C) kõvera lamedamaks (vajutades kõvera vasakut alguspunkti ja vedades seda üles), tähendab see seda, et väike värvi hulk tekitab suurema hulga värvi (kõrgem Y-väärtus). Sisuliselt tugevdad sa ähmaseid detaile ning vähendad samas üldist kontrasti. Saad ka tekitada eraldi punase värvi kõvera, mis võib näiteks luua tulemise, kus vähest punase hulka pole üldse näha, aga suurt hulka on.

Siin on mõned tavapärased kõverad, mille abil on lihtne saavutada soovitud efekti:



A) Helenda B) Negatiiv C) Vähenda kontrasti D) Posterda

Pöördväärtusta (*Invert*)

See sõlm tagastab sisendi pöördväärtuse.

Toon-küllastus-heledus (Hue Saturation Value)

Kasuta seda sõlme sisendpildi tooni, küllastuse ja heleduse muutmiseks.

Materjali vektorsõlmed (*Vector*)

Normaalisõlm (*Normal*)

Normaalisõlm arvutab normaalivektori ja skalaarkorrutise. Normaali suuna muutmiseks klõpsa keral ja lohista hiirt.

Selle sõlmega saad lisada uusi normaalivektoreid. Näiteks võid seda kasutada värvisegamissõlme (Color Mix) sisendina. Kasuta segamise teise sisendina pildisõlme Image. Tagajärjel tekkivat toonitud väljundit saab lihtsalt muuta valgusallika liigutamise teel (vajutades kerale ja lohistades seda).

(Külje) normaal tähistab külje suunda kaamera suhtes. Selle sõlmega saad saavutada järgnevat:

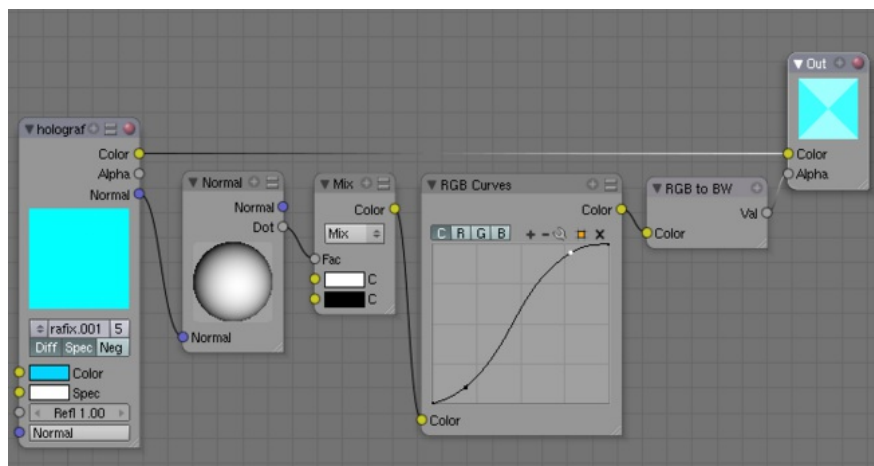
- Kasuta sõlme, et luua kindel suund -> väljund (Normal).
- Arvuta normaalisõlme Normal skalaarkorrutis Dot. Skalaarkorrutis Dot on ühemõõtmeline väärtus (number).
 - Kui kaks normaalit on sama suunaga, on skalaarkorrutis Dot 1.
 - Kui nad on risti, on skalaarkorrutis Dot null (0).
 - Kui nad on antiparalleelsed (nende suunad on vastupidised), on skalaarkorrutis Dot -1. *Ja arvasid, et ei kasuta enam kordagi neid teadmisi, mida said ülikooli vektorarvutuste klassist - häbi sulle!*

Seega saame teha igasugu erinevaid asju, mis sõltuvad vaatenurgast (nagu elektronmikroskoobi efekt või mõned neid tehnikatest, mida kirjeldati peatükis [Doc:Tutorials/Textures/Map_Input_Techniques](#)). Parim asi on see, et sa saad suunda interaktiivselt manipuleerida.

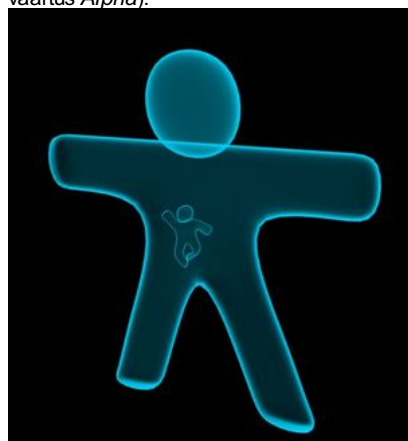
Üks hoiatus

Normaal arvutatakse külje, mitte piksli põhisel. Seega vajad sileda tulemise saamiseks palju külgi.

Normaalsõlme näidis



Skalaarkorrutise *Dot* kasutamine vaatenurgast sõltuva materjali loomiseks (antud juhul alfa väärtus *Alpha*).



Milline üllatus - Gus on rase!

Näidises kasutatakse skalaarkorrutist Dot materjali alfa väärtuse Alpha määramiseks. [RGB-kõverate](#) sõlme kasutatakse musta ja valge segunemisjoone teravdamiseks (mille tulemusena saavutatakse üleminek täiesti läbipaistvalt täiesti läbipaistmatule).

Materjaliks on tavaline sinine/helesinine materjal, millel on kõrge kiirgavuse väärtus Emit ja Z-läbipaistvus Z-Transp sisse lülitatud; taust on must. Seega: mida lähemale liigub külje nurk olekule, kus ta osutab otse kaamera suunas, seda läbipaistvamaks materjal muutub. Kui külj on suunatud kaamerast täisnurga all eemale (antud juhul suunaga üles või alla), muutub see läbipaistmatuks.

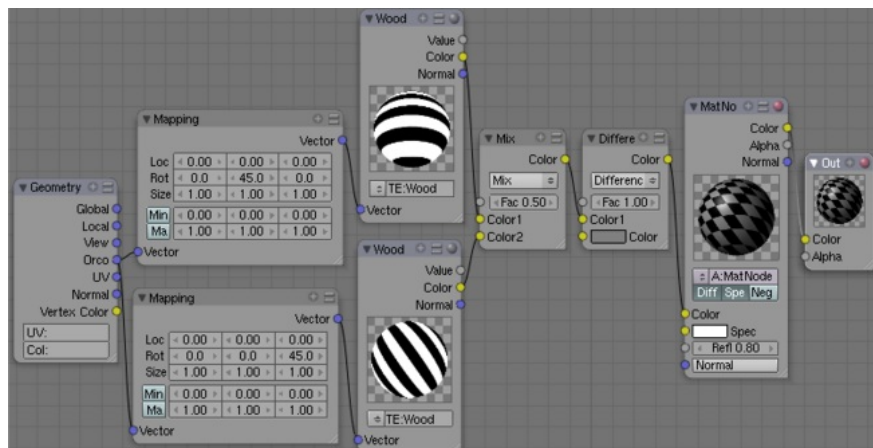
Normaali kasutamine mügarlike tekstuuride peegeldamiseks

Normaali saame kasutada peegelduse ja seeläbi ka läike muutmiseks, nagu on näidatud allpool. Seda efekti saab saavutada ka ilma sõlmedeta materjalide ja tekstuuripaneelil. Kuid sõlmi kasutades näed graafiliselt, mis lõpliku materjali loomisel toimub. Tulemuseks on see, et kuigi su mudeli pinnavõre on füüsiliselt sile, tekitab peegeldus tunde, et selle pinda katavad mügarikud või selle viimistlus on ebaühtlane (nagu oleks lisatud väga kehv värvikiht või halb pinnatöötlus). Realistlike pindade saavutamiseks on see väga oluline.

Põhimõtteliselt lubab laotamissõlm kasutajal laotust muuta. Sellega saad teha samu operatsioone kui materjali nuppude all oleva sisendi laotamise paneeliga Map Input. Samuti saad sellega teha mitut asja, mis tolle paneeliga on võimatu. Laotust saab soovi korral pöörata ja kinnitada. Antud hetkel toetab laotamissõlm aga ainult tasapinnalist laotamise tüüpi.

Sõlme sätted on reastatud X, Y, Z järjekorras. Kui soovid kasutada kinnitamise võimalust, proovi sisse lülitada miinimumi ja maksimumi nupud Min ja Max.

Laotamissõlme näidis



Laotamissõlmega imelise malelaua ruudustiku tekstuuri loomine.

Sellelt lihtsalt näitelt on näha ühte võimalikku laotamissõlme kasutamise võimalust huvitava tekstuuri loomiseks. Nagu näed, võib sama tekstuuri teistmoodi laotamine selle kasulikuks muuta. Sama tehnikat kasutades on lihtne jäljendada erinevaid triibulisi kangaid.

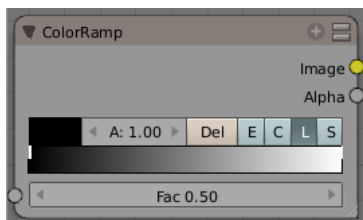
Vektorite kõverad (*Vector Curves*)

Selle sõlmega saad kõverate abil vektorväärtusi muuta.

Materjali konvertersõlmed (*Convertor Nodes*)

Nagu nimigi näitab, teisendavad järgnevad sõlmed mingil moel materjalide värve.

Värvülemineku sõlm (*ColorRamp*)



Värvülemineku sõlme kasutatakse kahe värvi vahelise värvisulatuse laotamiseks. See töötab samamoodi nagu [tekstuuride ja materjalide värvülemineku](#) ning kasutab faktori (*Factor*) väärtust näidatava ülemineku numbrivälja või indeksina ja saadab väljundist edasi värvi väärtuse ja alfa väärtuse.

Vaikimisi lisatakse värvülemineku sõlmele kahe vastanduva värviga. Täiesti must värv on vasakul (värvinäidises olev toon, mille alfa väärtus on 1.00) ja lubivalge paremal. Värv valimiseks vajuta LMB värvülemineku sees oleval kitsal vertikaalsel joonekesel. Näidispildil on valitud must värv, sest see on valgega esile tõstetud. Värv seaded on näidatud gradiendi kohal (vasakult paremale): värvi näidis, alfa väärtus ja interpolatsiooni tüüp.

Üleminekus valitud värvi tooni muutmiseks vajuta värvinäidisel LMB ja vali lahtihüppavast värvivalijast uus värv. Tooni kinnitamiseks vajuta **↵** Enter.

Värvide lisamiseks hoia all Ctrl ja klõpsa gradiendi sees LMB . Värvide muutmiseks vajuta kandiilisel värvikastil, mille peale hüppab lahti värvi muutmise dialoog. Alfa väärtuse muutmiseks lohista hallil numbriväljal. Kui ühendad alfa väljundi mõne RGB-värviseadega sisendiga *Factor*, saad kasutada teksture maskidena (või simuleerida varasemat kiirgamisfunktsiooni *Emit*).

Üleminekust mõne värvi kustutamiseks vali see ja vajuta kustutamisnuppu Delete.

Mitme värvi puhul saad interpolatsiooni valiku kaudu juhtida seda, kuidas nad ühest teise muunduvad. Kasuta interpolatsiooni nuppu määramaks, kuidas värvid moodustavad ülemineku: sujuvalt (*Ease*), kardinaalselt (*Cardinal*), lineaarselt (*Linear*) või kõverana (*Spline*).

Värvuspunktide läbipaistvuse määramiseks määra A: nupul väärtus.

Värvieemaldussõlm (*RGB to BW*)



See sõlm konverteerib RGB värvides pildi mustvalgeks.

Väljundi pesa ühendamine

Kui ühendad arväärtust (Val) väljastava väljundipesa mõne sisendipesaga, mis ootab sisendina pilti (Image), sisestab Blender nähtamatu värvülemineku sõlme ColorRamp, mis tõlgib väljundväärtuse materjali värviks.

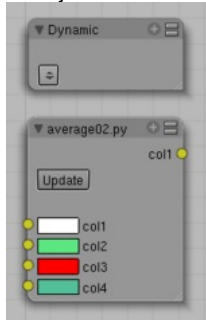
Matemaatika (*Math*)

Vektormatemaatika (*Vector Math*)

Väärtuse kokkusurumine (*Squeeze Value*)

Eralda RGB kanalid (*Separate RGB*)

Kombineeri RGB kanalid (*Combine RGB*)

Materjali dünaamilised sõlmed (*Dynamic Nodes, PyNodes*)

Dünaamilised sõlmed (*PyNode*). Ülemine on vaikeoleku näide. Alumisse on laetud skript.

Dünaamiliste sõlmede abil saad kirjutada oma enda erisõlmi. Need sõlmed kirjutatakse programmeerimiskeeles Python. Vaata täpsema info saamiseks [programmeerimisliidese osa](#).

Dünaamilise sõlme (*PyNode*) kasutamiseks lae soovitava sõlme skript Blenderi tekstiredaktorisse. Pärast seda pead lisama oma sõlmeskeemile dünaamilise sõlme. Laetud sõlmes on faili valimise nupp. Vali selle abil tekstiredaktorisse laetud fail. Pärast laadimist ehitatakse sulle vastav sõlm. Kui kood on töökorras, saad hakata sõlme kasutama. Kui kood ei tööta, otsi veateateid Blenderi konsoolist.

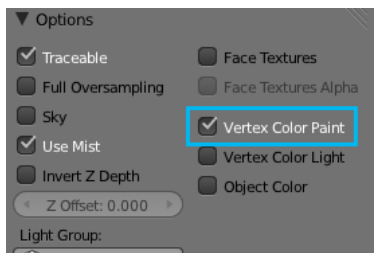
Pea meeles, et kui skripti muudad, pead vajutama dünaamilise sõlme värskendamise nuppu Update.

Dünaamiliste sõlmede retsepte leiad [retseptiraamatust](#).

Tippude maalimine (*Vertex Painting*)

Tippude (*vertex*) maalimine on lihtne moodus objektile värvide maalimiseks, selleks muudetakse tekstuuride kasutamise asemel otse tippude värve ning see on üsna lihtne.

Tippe saab maalida, lülitades sisse tippude maalimise režiimi (Vertex Paint Mode), kuid tulemust ei ole renderduses näha, kui sa ei pane materjalide paneelis linnukest kasti "Vertex Color Paint".



Sätted

Tööriista riulis (kiirvalik T) on enamus tippude maalimise tööriistu.

Pintsel (*Brush*)

Üleval servas oleva pildi pealt saad valid tööriista eelseadistusi, nimetada pintsleid ümber ning lisada ja kustutada spetsiaalselt loodud pintsleid.

Radius (raadius)

Määra pintslis raadius

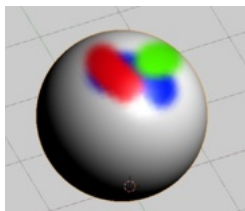
Strength (tugevus)

Määra pintslis efekti tugevus.

Tööriist (*Tool*)

Mix (segamine)

Segab RGB väärtuseid. Kui tugevuse väärtuseks on määratud 1.0, kaetakse alla jääv "värvikiht".



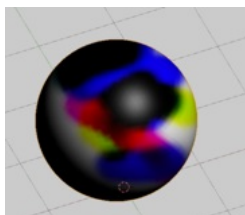
Täistugevusega segatud pealiskih

Add (liida)

Liidab RGB-väärtused. Muudab lõpuks terve objekti valgeks, kuna RGB-väärtused kasvavad järjest, kuni jõuavad väärtuseni 1.0-1.0-1.0: puhas valge.

Subtract (lahuta)

Lahutab RGB-väärtused. Tavaliselt on tulemuseks must.



Täistugevusega lahutamine

Multiply (korruta)

Korrutab pintslis värvid tippude värvidega.

Blur (hajuta)

Hajutab tippude värvid.

Lighten (helenda)

Muudab tippude värvi heledamaks.

Darken (tumenda)

Muudab tippude värvi tumedamaks.

Pintslitõmme (*Stroke*)

Airbrush (aerograaf)

Pintslit võimaldab värvi välja seni, kuni hiire nuppu hoitakse all, voolu tugevuse määrab kiiruse säte Rate. Kui see ei ole sisse lülitatud, muudab pintsel värvi ainult siis, kui selle asukoht muutub.

Smooth stroke (sujuv tõmme)

Pintsel jääb hiirest natuke maha ja tema liikumine on sujuvam. Sisselülitamisel muutuvad aktiivseks järgmised sätted:

Radius (raadius)

Määrab minimaalse kauguse eelmisest punktist, enne kui tõmme jätkub.

Factor (faktor)

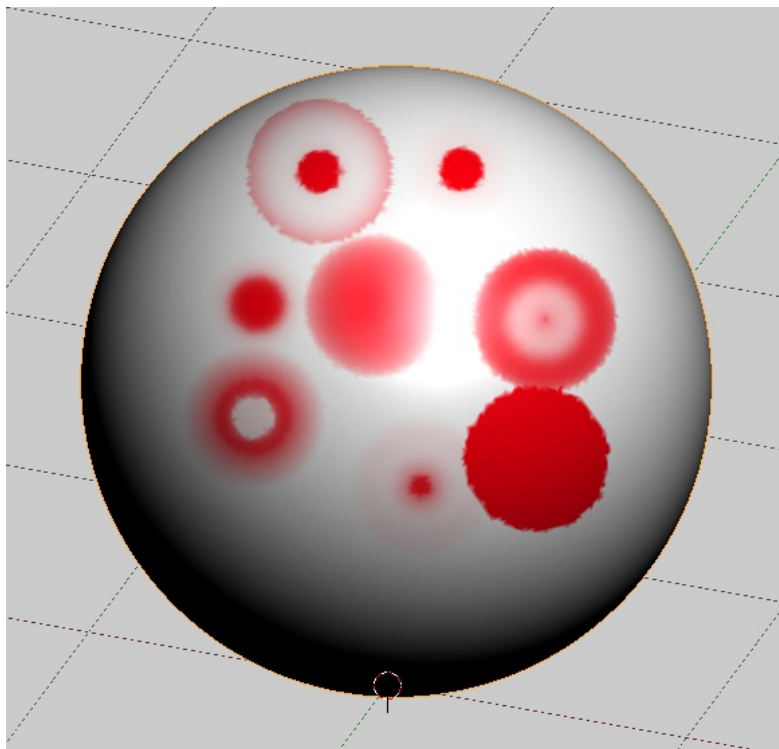
Määrab silumise määra.

Space (vahe)

Tekitab pintslitõmbeid punktide jadana, mille vahemaa määrab ära vahekauguse säte Spacing. Vahekaugus (Spacing) tähistab protsenti pintsli diameetrist.

Kõverad (Curves)

Pintslikõverad mõjutavad seda, kui tugevasti värvi lisatakse, sõltuvalt pintsli keskpunkti kaugusest. Teisisõnu saad sa nende läbi mõjutada pintsli intensiivsuse hajusust.

**Välimus (Appearance)**

Lubab sul muuta pintsli raadiuse piirjoone värvi ning määrata spetsiaalse ikooni.

Valikud (Options)**All Faces (kõik küljed)**

Värvib kõikidele pintsli raadiusse jäävatele külgedele. Kui see on välja lülitatud, värvitakse ainult pintsli keskpunktis olevat külge.

Normals (normaalid)

Määrab enne maalimist tipunormaali. See tavaliselt värvimist ei mõjuta.

Spray (pihusti)

See on vigane! Jätkab värvimist seni, kuni hiirt all hoitakse.

Üldised seaded**Size (suurus)**

Kõik pintslid kasutavad sama suurust.

Strength (tugevus)

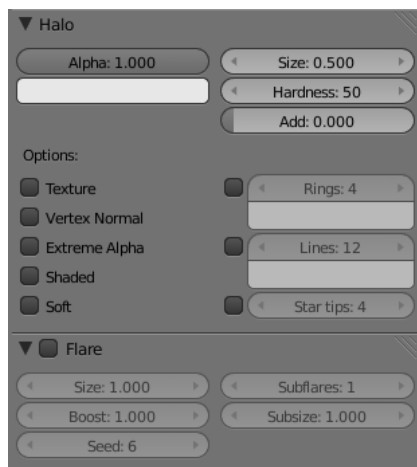
Kõik pintslid kasutavad sama tugevust.

Halomaterjalid

Blenderiga tulevad kaasa materjalid, mis ei allu külgedevarjutaja loogikale ning mida lisatakse tipu- ja mitte küljepõhiselt. Neid kutsutakse halodeks (Halos), sest nad on küll nähtavad, kuid neil puudub pind. Nad näevad välja nagu väikesed valguspilved, kuigi tegelikult nad ei ole üldse valgustid, sest ei lisa stseeni valgust nagu lambid.

Halod on kasulikud erinevate eriefektide loomisel, objekti helendama panekul või reaalse valgusti ümber nähtava valguse või udu/atmosfääri loomisel.

Seaded



Halo paneelid

Halode kasutamiseks vajuta Halo nuppu materjali konteksti (Material) ülemisel paneelil.

3D-vaates näed, et võre külgi enam ei renderdata. Selle asemel renderdatakse ainult tippe, sest need on kohad, kuhu iga üksik halo paigutub. Halosid võib olla keeruline segasest stseeni üles leida, seetõttu pane haloobjektile [Ülevaateakna](#) kaudu leidmiseks korralik nimi.

Seadistuste paneelis, kus tavaliselt asuvad hajuvuse (Diffuse), läike (Specular) ja varjutuse (Shading) paneelid, on nüüd näha paneelid, mis vastavad halo omadustele:

Halo paneel

Alpha (alfa)

Läbipaistvus

Color Swatch (värvivalik)

Halo enda värv

Size (suurus)

Määrab halo suuruse

HardnessTugevus ()

Määrab halo tugevuse. See sarnaneb läike tugevusega



Lisamise (Add) tulemus

Add (lisamine)

Lisamise numbriväli Add määrab selle, palju halo värvist 'lisatakse', mitte ei segata kokku selle taha jäävate objektide ning teiste halode värvidega. Lisamise suurendamisel näib halo valgustavat neid objekte, mis liiguvad läbi halo välja või selle taha.

Texture (tekstuur)

Annab halole tekstuuri. Vaikimisi lisatakse tekstuurid objektidele objektikoordinaatidega ning nad väljenduvad halos läbi selle värvuse muutuse, mis baseerub halo aluseks oleva tipu värvil. Selle valiku sisselülitamisel hakkab tekstuur tööle halo sees ja hakkab seeläbi muutma selle värve ja läbipaistvust; tekstuur laotatakse *igale* halole. See tehnika on väga kasulik osakeste süsteemide või millegi samase abil realistliku vihma efekti loomiseks.

Vertex Normal (tipu normaal)

Kasuta halo suuruse määramiseks tipu normaali

Extreme Alpha (üli-alfa)

Suurendab täiendavalt alfa väärtust

Shaded (varjutatud)

Halole langevad väliste objektide valgus ja varjud

Kui varjutamine on sisse lülitatud, mõjutab kohalik valgus halot; valgustid muudavad seda heldamaks ning mõjutavad selle hajuvuse värvi ja intensiivsust.

Soft (pehme)

Pehmendab halo servad nendes kohtades, kus need muu geomeetriaga kattuvad.

Efektid

Lülitab sisse kas osad või kõik efektid, määrab punktide/rõngaste arvu ja muudab neist iga individuaalset värvi:

Rings (rõngad)

Lisab halo ümber rõngad.

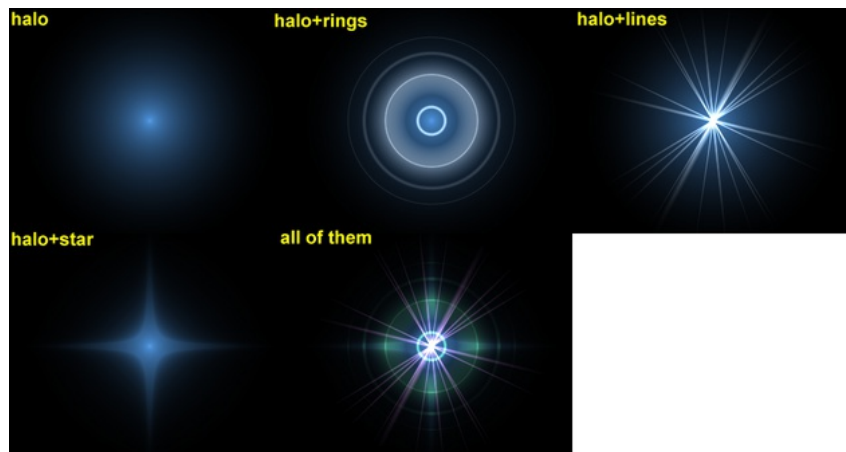
Lines (jooned)

Lisab halo keskpaigast algavad nooled.

Star tips (täheharud)

Annab halole tähe kuju.

Värviüleminekuid kasutada ei saa. Jooni, rõngaid ja mitmeid eriefekte saab lisada vastavate nuppudega, mille hulgas on objektiivi peegeldus (*Flare*), rõngad (*Rings*), jooned (*Lines*), täht (*Star*), tekstuur (*Texture*), üli-alfa (*Extreme Alpha*) ja varjutatud (*Shaded*). Pilt *Halo variatsioonid* näitab halo materjali lisamise tulemust ühest tipust koosnevale võrele.



Halo variatsioonid

Halo suurust, tugevust ja alfa saab muuta vastavate numbriväljade abil. Need on väga sarnased tavaliste materjalisätetega



Lisamise numbriväli Add määrab selle, palju halo värvist 'lisatakse' ja mitte ei segata kokku selle taha jäävate objektide ning teiste halode värvidega. Lisamise suurendamisel näib halo valgustavat neid objekte, mis liiguvad läbi halo välja või selle taha.

Rõngaste, joonte ja tähe harude arvu eraldi määramiseks tuleb pärast vastava valiknupu sisse lülitamise kasutada numbrinuppe rõngad (Rings:), jooned (Lines:) ja täht (Star:). Rõngad ja jooned asetatakse ja suunatakse juhuslikult, nende mustri muutmiseks tuleb muuta seemne numbrivälja Seed:, mis muudab juhuslikult genereeritud numbri alget.

Peegelduse paneel (*Flare*)

Peegelduse sisselülitamine renderdab halo objektiivi peegeldusena

Size (suurus)

Määrab faktori, mille võrra on peegeldus suurem kui halo

Boost (tõste)

Annab peegeldusele lisajõu

Seed (seeme)

Määrab peegelduse seemne tabeli nihke

Subflares (alampeegeldus)

Määrab alampeegelduste hulga

Subsize (alamsuurus)

Määrab alampeegelduste, -punktide ja -ringide suuruse

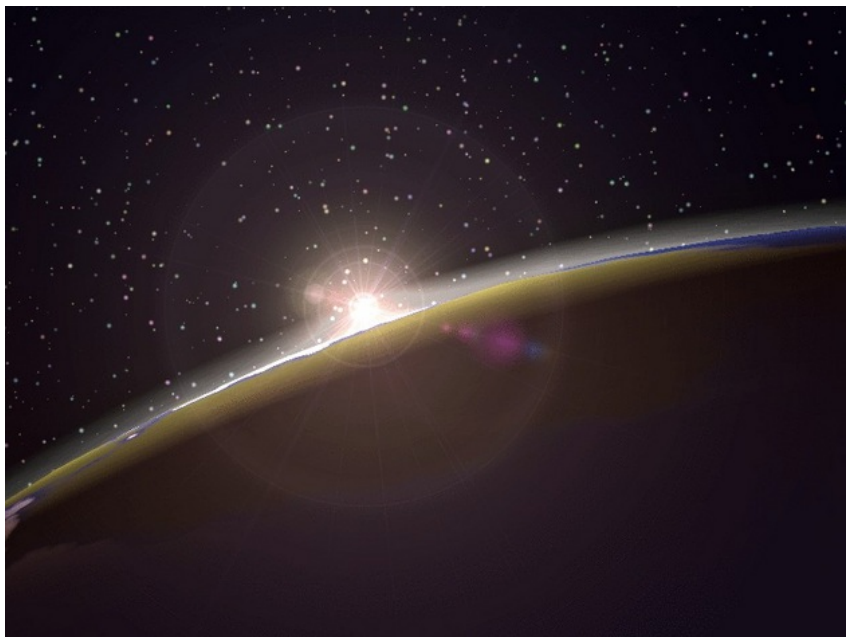
Objektiivi peegeldus

Meie silmad on sedasi treenitud, et nad usuvad, et pilt on reaalne, kui sellel on näha fotograafia mehhaaniliste protsesside artefaktid. Selliste artefaktide näidised on liikumishägu (*Motion blur*), sügavusteravus (*Depth of Field*) ja objektiivi peegeldused (*Lens Flares*). Esimest kahte kirjeldame *renderdamise peatükis*; viimast saab tekitada eriliste halodega. Simuleeritud objektiivi peegeldus ütleb vaatajale, et pilt on tehtud kaameraga, mis paneb teda uskuma, et see on reaalne.

Objektiivi peegeldusi saab Blenderis võreobjekti abil luua, kasutades esmalt materjali sätete Halo nuppu ja seejärel peegelduse valikut Flare varjutajate paneelis Shaders. Proovi lülitada sisse rõngad (Rings) ja jooned (Lines), kuid hoi a nende sätete värvid suhteliselt tagasihoidlikena. Mängi peegelduste arvu (Flares:) ja seemne sättega Fl.seed:, kuni jõuad silmale meeldiva tulemuseni. Vajaduse korral kasuta tugevama efekti saavutamiseks tõstet (Boost:) (pilt *Objektiivi peegelduse sätet*).

Pea meeles, et see ei simuleeri läbi klaasis läätse liikuvate footonite füüsikat, tegemist on lihtsalt iluasjaga.

Blenderi objektiivi peegeldus näeb liikudes ilus välja ning kaob, kui mõni teine objekt peegeldust tekitava võre varjab.



Objektiivi peegeldus

Halo tekstuur

Vaikimisi lisatakse tekstuurid objektidele objektikoordinaatidega ning nad väljenduvad halos läbi selle värvuse muutuse, mis baseerub halo baasiks oleva tipu värvil. Selle valiku sisselülitamiseks halo sees ja seeläbi halo värvide ja läbipaistvuse mõjutamiseks vajuta tekstuuri nuppu Texture; tekstuur laotatakse *igale* halole. See tehnika on väga kasulik osakeste süsteemide või millegi samase abil realistliku vihma efekti loomiseks.

Teine seade on varjutatud (*Shaded*). Kui varjutamine on sisse lülitatud, mõjutab kohalik valgus halot; valgustid muudavad seda heldamaks ning mõjutavad selle hajususe värvi ja intensiivsust.

Näited

Maatriksekraan

Kasutame halot maatriksekraani loomiseks.

- Alustamiseks lisa mõõtudega 32×16 ruudustik. Seejärel lisa kaamera ja muuda stseeni niimoodi, et sul oleks enne tehtud reklaamsildist kena vaade.
- Kasuta mõnd 2D pildiprogrammi ja tee lihtsa rasvase fondiga pilt punasest tekstist mustal taustal (kui oled laiskvorst, võid lihtsalt salvestada alloleva pildi oma kõvakettale). Pilt *Maatriksekraani pilditekstuur* on 512 pikslit lai ja 64 pikslit kõrge ning selle mõlemal küljel on natuke musta ruumi.



Maatriksekraani pilditekstuur

- Lisa reklaamtahvile materjal ning määra selle tüübiks Halo. Halo suuruseks (HaloSize) määra 0.06 ning stseeni renderdamisel peaksid sa nägema valgetest täppidest ruudustikku.
- Lisa tekstuur (Texture) ja mine seejärel tekstuuri nupule ja muuda see pilditekstuuriks. Pärast pildi laadimist uuesti renderdamisel peaksid nägema ruudustikus mõningaid punaseid punkte.
- Mine tagasi materjali nupule ning muuda suuruse parameeter (sizeX) umbes 0.5-ks ning renderda seejärel uuesti; tekst peaks nüüd olema reklaamtahvli keskel.
- Valgete punktide eemaldamiseks muuda materjali värv tumedaks ja renderda uuesti. Nüüd peaksid sul olema ainult punased täpid, kuid reklaamtahvel on ikka veel liiga tume. Selle parandamiseks sisene muutmisrežiimi ja kopeeri kõik tipud kiirvaliku ⇧ ShiftD abil (väldi nende liigutamist!). Seejärel muuda materjali alt nende lisamise väärtust Add.



Maatriksekraan.

Nüüd võid tekstuuri üle reklaamtahvli liikuma panna, kasutades tekstuuri paneeli (Texture) väärtust ofsX. (Võid kasutada suurema tihedusega ruudustikku, kuid sellisel juhul pead sa halode suurust vähendama, et need üksteisega ei kattuks). (*Maatriksekraan*).

Märkus materjali indeksite kohta

Halo materjalid töötavad ainult siis, kui nende määramisel kasutatakse esimest materjali indeksit. Järgmisel materjali indeksil olevaid materjale ei renderdata.

Tekstuuride sissejuhatus

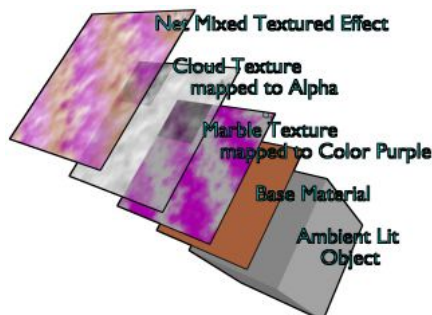
Arvutigraafikas nimetatakse tekstuurimiseks pindadele detailide lisamise meetodit, mille puhul pinnale projitseeritakse pildid või mustrid. Projitseeritud pildid ja mustrid võivad muuta mitte ainult pinna värvi, vaid ka läiget, peegeldust, läbipaistvust või isegi simuleerida kolmemõõtmelist sügavust. Tavaliselt projitseeritakse pildid ja mustrid renderdamiseks, kuid tekstuurimist kasutatakse ka voolimisel, maalimisel ja objektide deformeerimisel.

Blenderis saab teksture:

- lisada materjalidele (*Material*)
- lisad [maailma tagaplaanile](#).
- lisada pintslitele (*Brush*), näiteks:
 - [voolimisrežiimis](#)
 - [tekstuuride maalimisel](#)
- kasutada koos töötlejatega Modifiers, näiteks:
 - osakeste tekstuurid
 - ookeani tekstuurid

Materjalitekstuurid

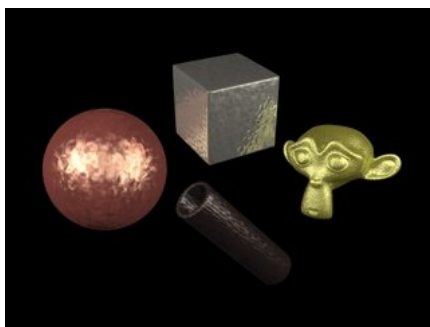
Senivaadeldud materjaliparameetrite abil saab luua siledaid, *ühtlasi* objekte, kuid sellised objektid ei ole eriti realistlikud. Reaalses maailmas on ühtlane struktuur pigem erand ning mõjub kohalt. Et taolisest ebarealistlikust ühetaolisusest üle saada, võimaldab Blender kasutada *teksture*, mis muudavad materjalide peegelduvust, läiget, karedust ja teisi pinna omadusi.



Tekstuurikiht baasmaterjalil

Tekstuurid käituvad nagu lisakihid baasmaterjali peal. Nad mõjutavad objekti summaarse värvuse üht või mitut aspekti. Objekti lõplik värvus näitepildil moodustub üksteisele järgnevate üksikefektide summast. Need üksikud kihid on:

1. Objekti valgustav **üldvalgus** (*Ambient Light*) vastavalt maailma seadistustele.
2. Sinu **baasmaterjal** katab kogu pinna ühtlase värviga, mis koosmõjus valgusega tekitab erinevad hajus- (*Diffuse*), läike- (*Specular*) ja peegelduse (*Mirror*) värvid vastavalt sellele, kuidas valgus tabab pinda, läbib seda ja peegeldub sellelt.
3. Meil on **baastekstuuri** kiht, mis lisab baasmaterjali peale purpurse marmorimustri.
4. Järgmisena on meil **teine pilvetekstuur**, mis läbipaistvuse (*Alpha*) väärtust kasutades muudab pinna udusalt/hajusalt läbipaistvaks.
5. Need kaks tekstuuri **segatakse** baasmaterjaliga kokku summaarseks efektiks, purpurpruuni uduga kuubiks.



Mõned metallitekstuurid

Võimalus kasutada summaarse efekti loomiseks *rohkem kui ühte* tekstuuri on üks Blenderis realistliku väljanägemisega objektide loomise "saladus". Kui sa suvalist objekti tegelikult maailmas uurides vaatad tähelepanelikult "valgust ja varju", märkad sa, et objekti lõplik väljanägemine moodustub paljudest iseseisvatest visuaalsetest omadustest, mis kombineeruvad omavahel erinevatel viisidel ja erineval määral. Need omadused võivad sõltuvalt vaatenurgast, valgustingimustest jms olla rohkemal või vähemal määral ilmsed. Blender võimaldab sul seda efekti saavutada mitmel viisil. Sa saad kasutada "tekstuurikihtide pinu", nagu kirjeldatakse [selles peatükis](#). Samuti võid sa koostada "tekstuurisõlmedest" suvalise keerukusega võrgustikke ("nuudleid"), nagu kirjeldatakse [siin](#). Valik on sinu.

Tekstuurid jagatakse kolme põhikategooriasse:

Protseduuriilsed tekstuurid (Procedural Textures)

Tekstuurid, mis genereeritakse matemaatiliste funktsioonide abil. Näiteks puit (Wood), pilved (Clouds) ja moonutatud müra (Distorted Noise)

Pildid ja filmid (Images or Movies)

Fotod ja filmid, mis projitseeritakse objektidele. Näiteks kerale laotatud tasapinnaline maailmakaart.

Keskonnakaardid (Environment Maps)

Tekstuurid, mida kasutatakse valguse peegeldumise ja murdumise mulje loomiseks. Näiteks tänava pilt, mis peegeldub autoaknalt.

Täpsema ülevaate saamiseks võid sa lugeda õppetükki [Tekstuuride kasutamine](#).

Maailma tekstuurid (*World*)

Vaja teha

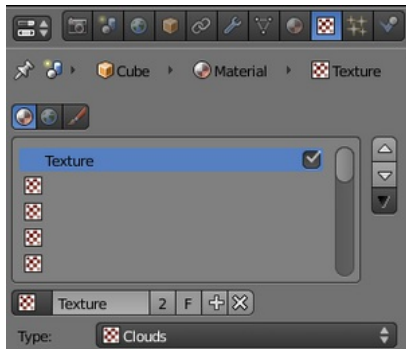
Pintsli tekstuurid (*Brush*)

Vaja teha

Tekstuuri määramine

Sellel leheküljel seletatakse lihtsalt, kuidas pesasse tekstuuuri lisada. Tekstuuride üldiseid omadusi seletatakse lahti [siin](#).

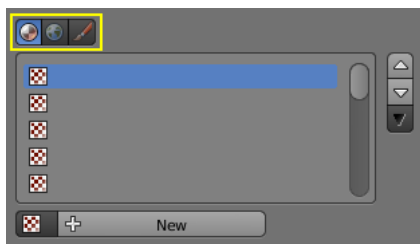
Tekstuuri konteksti valimine



Tekstuuri paneel Texture

Vali omaduste redaktoris (*Properties editor*) tekstuuuri kontekst *Texture*: see näitab sulle tekstuuuri paneeli *Texture*.

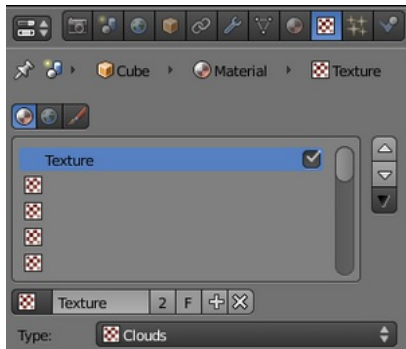
Tekstuuri andmetüübi valimine



Tekstuuri paneel Texture, milles on nupud materjal (Material), maailm (World) ja pintsel (Brush) esile tõstetud

Tekstuuri paneeli ülaotsas olevad nupud materjal (Material), maailm (World) ja pintsel (Brush) näitavad tekstuuuri andmetüüpi (*Texture data type*) ehk siis seda, millist sorti tekstuuuri muudetakse.

Tekstuuri pesad



Tekstuuri paneel Texture

Nende nuppude all olev nimekiri kujutab meie poolt hallatavat tekstuuride pinu (Stack). Selles võib olla kuni kaheksateist tekstuuuri pesa (Texture Slots):

- Tähista tekstuur linnukesega, et seda sisse lülitada (või eemalda välja lülitamiseks linnuke).
- Kasuta paremal pool olevat kolme nuppu, et üksikut tekstuuuri pinus üles-alla liigutada või materjali sätteid pesade vahel kopeerida ja kleepida.

Uude tekstuuuri pesasse uue tekstuuuri andmebloki loomine

Vali tühi pesa ja vajuta seejärel nuppu **+ New**.

See teeb kahte asja:

- loob uue tekstuuuri andmebloki
- ja lisab tekstuuride pinusse uue pesa


Täis pesasse uue tekstuuuri andmebloki loomine

Vali mõni täis pesa ja vajuta seejärel nuppu **+**.

See teeb kahte asja:

- loob uue nimega uue tekstuuri andmebloki, **tehes valitud pesasse määratud tekstuuri andmeblokist koopia**
- määrab selle uue andmebloki asukohaks valitud pesa

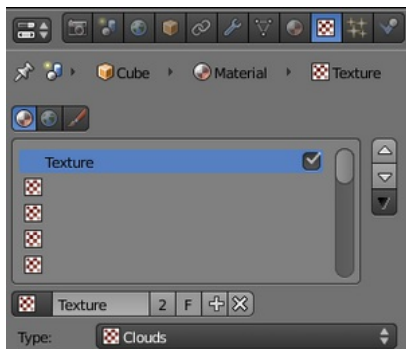
Tekstuuri andmebloki jagamine täis pesaga

- Vali täis pesa ja vajuta sirvimise nuppu . See avab menüü, kus on ära toodud kõik avatud failis olevad tekstuuri andmeblokid.
- Vali menüüs mõni tekstuuri andmeblokk, et määrata selle asukohaks valitud pesa. See jagab valitud tekstuuri rohkem kui ühe objektiga ning seetõttu tõuseb andmeblokis näidatav kasutajate arv (*Number of users*) ühe võrra.

Tekstuuri üldised sätted

Omaduste muutjas (*Properties editor*) vali tekstuuri kontekst *Texture*. Seejärel näed tekstuuri paneeli *Texture*.

Tekstuuri pinu



Tekstuuripaneel Texture

Nende nuppude all olevad nupud näitavad tekstuuride pinu (Stack), mida saame kasutada. Siin saab olla kuni kaheksateist tekstuurikanalit (Texture Slots):

- Tekstuuri saab vastavas materjali kanalis sisse ja välja lülitada, märkides või tühistades linnukese tekstuurivälja taga.
- Kolm nuppu paremas servas võimaldavad üksikuid tekstureid pinus üles ja alla liigutada ning materjali omadusi ühelt väljalt teisele kopeerida.

Tekstuuri andmeblokk

Vali tekstuuride pinu lahter, et näha selle sätteid.

Esimene nuppude grupp pinu all näitab parajasti pinus valitud tekstuuri.

Browse (sirvi)

Esimene nupp pinu all võimaldab valida tekstuuri kõigi hetkel avatud failis leiduvate hulgast. Tekstuurid on terves failis ühised ja ühte tekstuuri saab seostada rohkem kui ühe materjaliga. Kui sa oled tekstuuri juba loonud ja tahad seda uuesti kasutada, vali see siit nimekirjast.

Name (nimi)

Tekstiväli, kus saab määrata materjali nime.

Number of Users (kasutajate arv)

Kui valitud tekstuur on kasutusel veel teises materjalis, on nupu peal 2. Siia vajutades saab vajadusel teha materjalist ühe kasutajaga koopia. Nii saad kiiresti luua uue tekstuuri juba olemasoleva põhjal.

Fake User (võltskasutaja)

Nupp F lisab valitud tekstuurile "libamaterjali", nii et see tekstuur salvestatakse faili isegi siis, kui tal ei ole "tõelisi" kasutajaid.

Add (lisa)

Asendab hetkel aktiivse tekstuurivälja uue loodud tekstuuriga.

Unlink (eemalda)

Eemaldab tekstuuri aktiivselt väljalt.

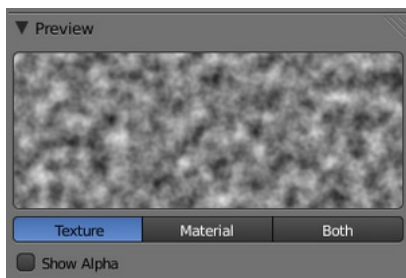
Tekstuuri tüüp

Vali tekstuuri tüüp, mida kasutada hetkel valitud tekstuuribloki jaoks. Tekstuuride tüüpe kirjeldatakse järgmises peatükis. Igal tüübil on oma spetsiifilised valikud.

- [Protseduuriilised tekstuurid \(Procedural Textures\)](#):
- [Pildi- ja videotekstuurid \(Image/Video Textures\)](#)
- [Keskonnapiildid \(Environment Map\)](#)
- [Mahulised tekstuurid \(Volume Textures\)](#)
- [Ookeani tekstuurid \(Ocean Textures\)](#)

Neid tüüpe kirjeldatakse [selles peatükis](#).

Eelvaade (Preview)



Eelvaate paneel Preview

Sellel paneelil näed kiiret eelvaadet, kuidas antud tekstuur üksikult, ilma objektile laotamata välja näeb.

Texture (tekstuur), Material (materjal) või Both (mõlemad)

Võimaldab valida kas ainult teksturi, ainult vastava materjali või korraga mõlema näitamise.

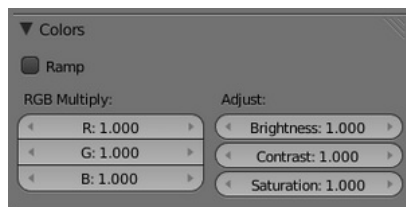
Show Alpha (näita läbipaistvust)

Näitab eelvaates ka läbipaistvust (*Alpha*).

Kui *Alpha: Use* (läbipaistvus: kasuta) on [pildilugemise](#) paneelil *Image Sampling* sisse lülitatud, siis näidatakse pildi läbipaistvust (alfakanalit).

Kui *Alpha: Use* (läbipaistvus: kasuta) on välja lülitatud, siis arvutatakse eelvaatepildil läbipaistvust samal meetodil nagu [mõjupaneelil Influence](#) - R, G ja B kanali keskmise järgi.

Värvid (Colors)



Värvide paneel Colors

Väli Ramp (värvi üleminek) aktiveerib seadistused, mis võimaldavad teisendada tekstuurivärve uuteks värvideks. Seda, kuidas neid üleminekuid kasutada, vaata [üleminekute](#) peatükist.

Tekstuuri värvi saab muuta nuppude Brightness (heledus), Contrast (kontrast) ja Saturation (küllastatus) abil. Kõiki RGB kanalitega tekstuure —, sealhulgas pilte (Images) ja keskkonnapilte (Environment Maps) — saab muuta ka RGB väljade abil.

R, G, B

toonivad teksturi muutes vastavalt punast, rohelist ja sinist kanalit intensiivsemaks.

Brightness (heledus)

muudab teksturi üldist heledust/intensiivsust

Contrast (kontrast)

muudab teksturi kontrasti

Saturation (küllastatus)

muudab teksturi küllastatust

Laotus (Mapping)

Selle paneeli valikud määravad, kuidas tekstuur pinnale laotatakse.

Pintslid

Need sätted ei ole pintslite puhul kasutatavad, sest sellel poleks mõtet

Vaata täpsemaid detaile [laotuse](#) peatükist.

Mõju (Influence)

Siit saad määrata, milliseid materjali omadusi tekstuur mõjutab ja kui palju.

Täpsem kirjeldus on [mõju](#) peatükis.

Pintslid

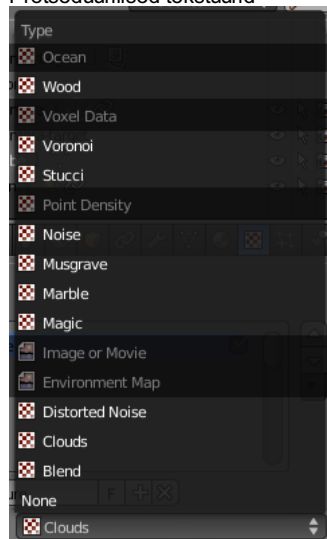
Need sätted ei ole pintslite puhul kasutatavad, sest sellel poleks mõtet

Tekstuuri tüübid

Kasutatavad tekstuuride tüübid on järgnevad:

- [Protseduurilised tekstuurid](#)
- [Pilditekstuurid](#)
- [Videotekstuurid](#)
- [Tekstuuri sõlmed](#)
- [Mahuga tekstuurid](#)
- [Ookeani tekstuurid](#)

Protseduurilised tekstuurid



Tekstuuritüüpide nimekiri (Texture Type) tekstuurinuppude tekstuuripaneelil.

(Mitteprotseduurilised tekstuurid on muudetud tumedamaks.)

Protseduurilised tekstuurid on sellised tekstuurid, mille muster on genereeritakse matemaatiliselt. Neid on üldiselt lihtne kasutada, sest neid ei pea spetsiaalselt võrele laotama, mis aga ei tähenda, et protseduurilised tekstuurid ei võiks olla vägagi kompleksed.

Need tekstuuritüübid on 'tõeliselt' kolmemõõtmelised. See tähendab, et nad sobivad servadel veatult kokku ja näevad välja, nagu vaja, ka siis, kui neid lõigata – nagu oleks tükk puud tööpoolest kaheks osaks lõigatud. Protseduurilisi tekstuure ei filtreerita ja neil ei siluta särke. See ei ole üldiselt ka probleem, sest kasutaja saab kergesti hoida genereerimise sagedused sobivates piirides.

Võimalikud tüübid on:

- [Krohv](#)
- [Maagia](#)
- [Marmor](#)
- [Moonutatud müra](#)
- [Musgrave](#)
- [Müra](#)
- [Pilved](#)
- [Puit](#)
- [Sulandamine](#)
- [Voronoi](#)

Ühised omadused

Müra baas

Kõigil müral põhinevatel Blenderi tekstuuridel (välja arvatud Voronoi ja lihtne müra) saab valikuga Noise Basis (müra baas) määrata põhialgoritmi, mille abil tekstuur genereeritakse. Selles nimekirjas on ka algne Blenderi enda müraalgoritm. Võimalus müra baasi määrata teeb protseduurilised tekstuurid erakordselt paindlikuks (eriti Musgrave).

Müra baas määrab tekstuuri väljanägemise põhistruktuuri (Blender 2.44 galerii):

Rakuline (*Cellnoise*)

Praguline Voronoi
(*Voronoi Crackle*)

Voronoi F2-F1

Voronoi F4

Voronoi F3

Voronoi F2

Voronoi F1

Blenderi algne
(*Blender Original*)

Müra baasil on lisaks veel kaks võimalikku väärtust, mis on väga sarnased Blenderi algsele (Blender Original):

- Täiustatud Perlin (*Improved Perlin*)
- Algne Perlin (*Original Perlin*)

Nabla

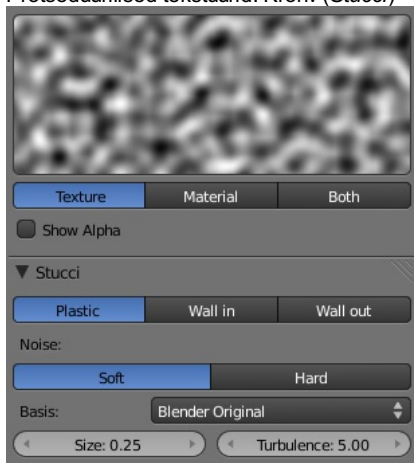
Peaaegu kõik Blenderi protseduurilised tekstuurid (välja arvatud sulandamine (Blend) ja maagia (Magic) kasutavad normaalide arvutamiseks tuletisi. See on oluline normaali- ja nihkekaartide jaoks. Nabla väärtus määrab tuletise mõju suuruse.

Soovitused

Kasuta suuruse (*Size*) nuppe laotuse paneelil Mapping määramaks, kui suurele alale protseduurilised tekstuurid laotatakse.

Protseduurilised tekstuurid võivad genereerida kas värvilisi, ainult intensiivsuse, läbipaistvusega või normaalitekstuure. Kui kasutada intensiivsustekstuuri, on tulemus mustvalge pilt, mida saab märgatavalt muuta, kasutades värvüleminekuid (*Ramp*). Teisest küljest aga, kui sa kasutad värvüleminekuid ja vajad ainult intensiivsuse väärtust, siis tuleb sisse lülitada nupp No RGB (mitte RGB) laotuspaneelil Mapping.

Protseduurilised tekstuurid: Krohv (*Stucci*)



Krohvitekstuuri paneel Stucci

Krohvitekstuur Stucci põhineb mürafunktsioonidel.

Sagedased kasutusala

Kivi, asfalt, apelsinid. Tavaliselt kasutatakse faktuurikaardina teraliste pindade saamiseks.

Väljund(id)

Normaalid ja intensiivsus

Valikud

Plastic/Wall In/Wall out (plast/sein sisse/sein välja)

Plast on selle tekstuuri põhivorm, seevastu "seinad" andsid talle nime (*stucci* ehk krohv). See on tavaline kiviseina muster aukude või konarustega.

Soft/Hard (pehme/terav)

Kaks võimalikku mürafunktsiooni varianti

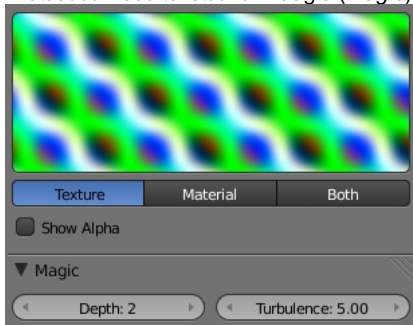
Size (suurus)

Müratabeli suurus

Turbulence (turbulents)

Krohvitekstuuri arvutussügavus

Protseduurilised tekstuurid: Maagia (*Magic*)



Maagiatekstuuri paneel Magic

Sagedased kasutusala

Ei kasutata kuigi sageli. Seda saab kasutada "õhukese kihi interferentsi" loomiseks, kui määrata laotuseks (Mapping) peegeldus (Reflection) ja kasutada suhteliselt suurt turbulentsi (Turbulence) väärtust.

Väljund(id)

RGB värvid. R, G ja B komponendid genereeritakse eraldi siinusfunktsiooni järgi.

Valikud

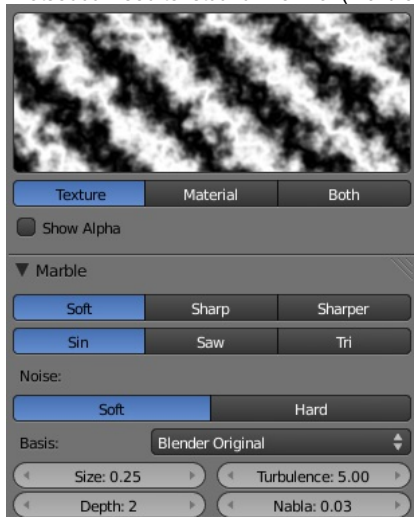
Depth (sügavus)

Arvutussügavus. Suur number pikendab oluliselt arvutusaega, kuid suurendab detailsust.

Turbulence (turbulents)

Mustri tugevus.

Protseduurilised tekstuurid: Marmor (*Marble*)



Marmortekstuuri paneel Marble

Sagedased kasutusala

Marmor, tuli, struktureeritud müra

Väljund(id)

Ainult intensiivsus

Ribad moodustuvad kas siinus-, saehammas- või kolmnurklainetest ja müra poolt määratud turbulentsist.

Valikud

Soft/Sharp/Sharper (sile, terav, teravam)

Kolm eelhäälestust udusest kuni selge marmorimustrini

Sin/Saw/Tri (siinus, saehammas, kolmnurk)

Triipe moodustavate lainete kuju

Soft/Hard (pehme/terav)

Kaks võimalikku mürafunktsiooni varianti.

Size (suurus)

Müratabeli suurus

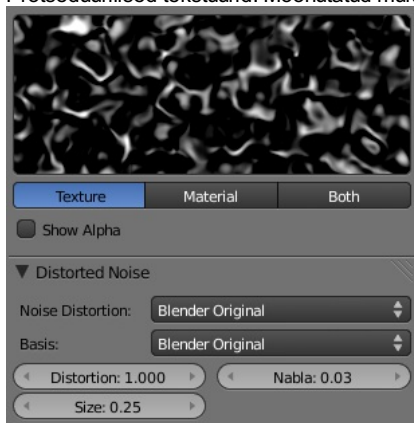
Depth (sügavus)

Marmori arvutussügavus. Suur number pikendab oluliselt arvutusaega, kuid suurendab detailsust.

Turbulence (turbulents)

Ribade turbulents (keerised ja ümberpaiknemised).

Protseduurilised tekstuurid: Moonutatud müra (*Distorted Noise*)



Moonutatud müra paneel Distorted Noise

Sagedased kasutusala

Grunge, väga kompleksne ja võimalusterohke

Väljund(id)

Intensiivsus

Valikud

Noise Distortion (müra moonutus)

Tekstuur, kust loetakse moonutus

Basis (baas)

Tekstuur, mida moonutatakse

Noise (müra)

Genereritud müra suurus

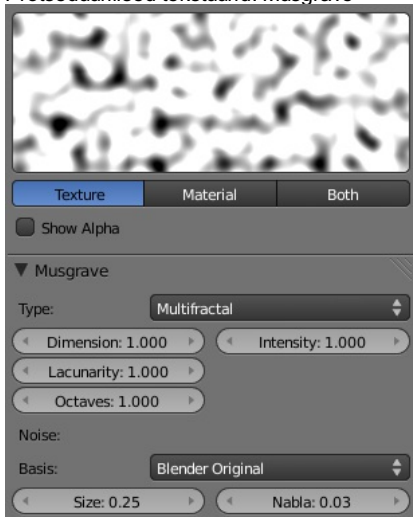
Distortion

Kui palju müra moonutus mõjutab baasi

Tehnilised detailid

Müramoonutus loeb väärtused sinu valitud baasist ning filtreerib neid, moodustades hübriidse mustri.

Protseduurilised tekstuurid: Musgrave



Musgrave teksturi paneel

Sagedased kasutuslald

Looduslikud materjalid – kuid ta on väga paindlik. Sa saad temaga teha peaaegu ükskõik mida.

Väljund(id)

Intensiivsus

Valikud

Type (tüüp)

Sellel protseduurilisel materjalil on mustri genereerimiseks võimalik kasutada viit erinevat müra tüüpi, mida saab vahetada menüüst paneeli ülaosas. Need viis tüüpi on:

- Hetero Terrain
- fBm
- Hybrid Multifractal
- Ridged Multifractal
- Multifractal

Need müra tüübid määravad viisi, kuidas Blender laotab järjestikuseid sama mustri koopiasid erineva kontrastsuse ja suurusega üksteise peale.

Põhilistel müratüüpidel on neli parameetrit:

Dimension (dimensioon)

Fraktaali dimensioon määrab kihi suhtelise kontrastsuse võrreldes eelmise kihiga. Mida suurem on fraktaali dimensioon, seda kõrgem on kontrast iga kahe kihi vahel ja seda rohkem detaile tekstuuril on. Vahemik: 0-st 2-ni.

Lacunarity (lakunaarsus)

See skaleerib Musgrave'i teksturi igat järgnevat kihti, st iga järgneva kihi mõõtkava on selle numbri pöördväärtus. St Lakunaarsus = 2 -> Mõõtkava = 1/2 algsest. Vahemik: 0-st 6-ni.

Octaves (oktaavid)

Määrab, mitu korda algne müramuster iseendale lisatakse (sealjuures iga kord mõõtkava ja kontrasti vastavalt dimensiooni ja lakunaarsuse väärtuste järgi teisendades). Vahemik: 0-st 8-ni.

Intensity (intensiivsus)

Valguse intensiivsus. Müratüübi Hetero Terrain korral nimetatakse seda nihkeks (Offset). Vahemik: 0-st 10-ni.

Müratüüpidel Hybrid Multifractal ja Ridged Multifractal on veel järgmised valikud:

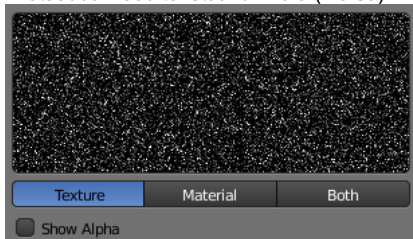
Offset (nihe)

Mõlemal on "fraktaali nihke" (*Fractal Offset*) nupp, mis töötab "merepinna" kõrguse määrajana ja muudab genereeritava faktuuri baaskõrgust. Sellest väärtusest madalamate kühmude kõrguseks pannakse 0. Vahemik: 0-st 6-ni.

Gain (tõus)

Määrab selle funktsiooni poolt genereeritud väärtuste vahemiku. Mida suurem on number, seda laiem on vahemik. Sellega saab tekstuurile, mille äärmised väärtused on maha lõigatud, kiiresti detaile lisada. Vahemik: 0-st 6-ni.

Protseduurilised tekstuurid: Mära (*Noise*)



Müratekstuuri paneel Noise

Sagedased kasutusala

Valge müra animatsioonis. See üldjuhul ei sobi, kui sa ei kasuta animatsiooni. Materjali ebaühtluse saavutamiseks kasuta pilvematerjali.

Väljund(id)

Intensiivsus

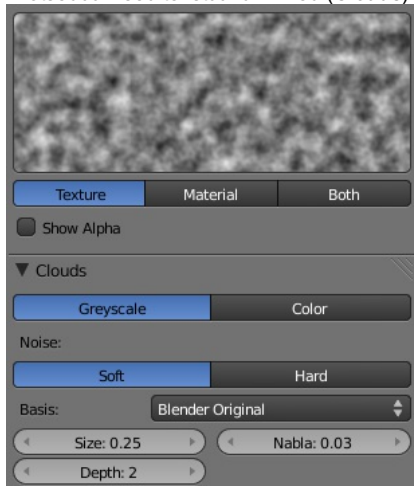
Valikud

Sellel meetodil ei ole paneeli ega nuppe. Lihtsalt lülita ta sisse.

Tehnilised detailid

Ehkki see näeb hea välja, ei ole ta Perlini müra! See on tõeline, juhuslikult genereeritud müra. Iga kord, igas kaadris, iga piksli jaoks annab ta erineva tulemuse.

Protseduurilised tekstuurid: Pilved (*Clouds*)



Pilvetekstuuri paneel Clouds

Sagedased kasutusala

Pilved, tuli, suits. Sobib hästi faktuurikaardiks, andes materjali pinnale üldise ebaühtluse.

Väljund(id)

Halltoonid (Greyscale, vaikimisi) või RGB värvid (Color)

Valikud

Greyscale (halltoonid)

Tavaline müra, määrab intensiivsuse

Color (värvid)

Müra määrab RGB väärtuse

Noise (müra)

Soft (pehme) või Hard (terav), muudab kontrasti ja teravust

Size (suurus)

Müratabeli suurus

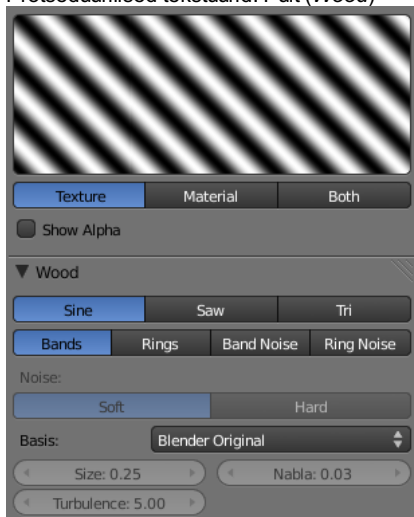
Depth (sügavus)

Pilvede arvutussügavus. Suur number pikendab oluliselt arvutusaega, kuid suurendab detailsust.

Tehnilised detailid

Tekstuuri genereerimiseks kasutatakse kolmemõõtmelist pseudojuhuslike arvude tabelit, millest iga 3D-punkti jaoks loetakse interpolateeritud parameetri väärtus (tänu Ken Perlinile meisterliku artikli eest: "An Image Synthesizer", SIGGRAPH proceedings 1985). Seda arvutusmeetodit kutsutakse ka Perlini müra. Lisaks on kõigil müral põhinevatel Blenderi tekstuuridel (välja arvatud Voronoi ja lihtne müra) uus valik Noise Basis (müra baas), mille abil saab määrata tekstuuuri genereerimise algoritmi.

Protseduurilised tekstuurid: Puit (*Wood*)



Puidutekstuuri paneel Wood

Sagedased kasutusala

Puit ja ringikujulised mustrid.

Väljund(id)

Ainult intensiivsus

Valikud

Sin/Saw/Tri (siinus, saehammas, kolmnurk)

Triipe moodustavate lainete kuju

Bands/Rings/Band Noise/Ring Noise (triibud/rõngad/triibumüra/rõngamüra)

Määrab, kas triibud on sirged või rõngakujulised, müraga või ilma

Soft/Hard (pehme/terav)

Kaks võimalikku mürafunktsiooni varianti

Size (suurus)

Müratabeli suurus

Turbulence (turbulents)

Müra turbulents triibu- ja rõngasmüra jaoks

Tehnilised detailid

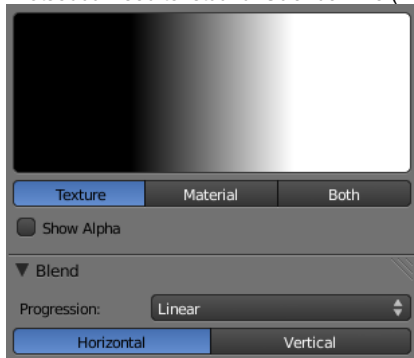
Genereerimine

Triibud genereeritakse siinusfunktsioonist. Sellele saab lisada mürafunktsiooni poolt genereeritud turbulentsi.

Koordinaadid

Et triibud genereeritakse siinusfunktsiooni abil, kordub tekstuur iga pi ühiku, mitte 1.0 ühiku tagant. Et seda parandada, muuda soovi korral tekstuuri mõõtkava pi väärtuse võrra.

Protseduurilised tekstuurid: Sulandamine (*Blend*)



Sulandamistekstuuri paneel Blend

Sagedased kasutusala

See on üks kõige sagedamini kasutatav protseduuriline tekstuur. Sulandamistekstuuri saab kasutada teise tekstuuride kokkusegamiseks (matriitsi (Stencil) abil) või kenade efektide tegemiseks (eriti kui kasutada normaalidele laotamise trikki). Pea ainult meeles, et kui sa kasutad sulandamise täpsemaks määramiseks värviüleminekut (*Ramp*), pead sa juhul, kui seostusmeetod Map to vajab intensiivsussisendit, lülitama sisse ka No RGB (mitte RGB).

Väljund(id)

Intensiivsus. Värviüleminek loob sujuva interpoleeritud progressiooni.

Valikud

Progression (gradient)

Sulandamise gradient

Linear (lineaarne)

Lineaarne gradient

Quadratic (ruutfunktsioon)

Ruutfunktsiooni gradient

Easing (rahnemine)

Sujuv, mittelineaarne gradient

Diagonal (diagonaalne)

Diagonaalne gradient

Spherical (kerakujuline)

Kolmemõõtmelise kuuli kujuline gradient

Quadratic Sphere (ruutfunktsiooniga kera)

Kolmemõõtmelise kuuli kujuline ruutfunktsiooni gradient

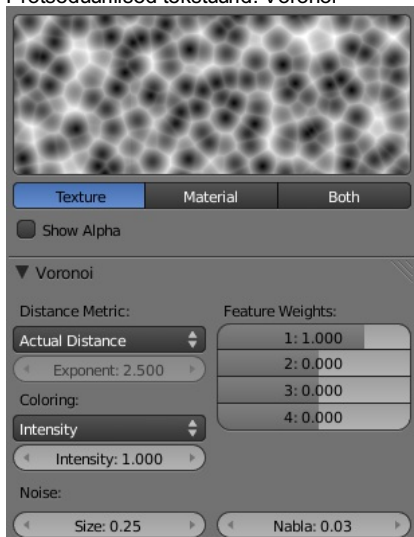
Radial (radiaalne)

Radiaalne gradient

Horizontal/Vertical (horisontaalne/vertikaalne)

Gradiendi suunda pööratakse täisnurga võrra.

Protseduurilised tekstuurid: Voronoi



Voronoi tekstuuri paneel

Sagedased kasutusala

Väga tõetruu metall, eriti "vasardatud" efektiga. Looduslikud materjalid (näiteks soomused ja veresooned naha all).

Väljund(id)

Intensiivsus (vaikimisi) ja värv

Valikud

Distance Metric (kauguse algoritm)

Sellel tekstuuril on palju kaugusalgoritmi variante. Need määravad, kuidas arvutatakse kaugus tekstuuri rakkude vahel. Need valikud on:

- Minkovsky
- Minkovsky 4
- Minkovsky 1/2
- Chebychev
- Manhattan
- Distance Squared (vahemaa ruut)
- Actual Distance (tegelik kaugus)

Valikul Minkovsky on kasutaja poolt määratav lisavalik (nupp Exponent (eksponent)), mis määrab Minkovsky kaugusfunktsiooni $(x^e + y^e + z^e)^{1/e}$ eksponenti (e). Väärtus 1 annab tulemuseks Manhattani kaugusfunktsiooni, ühest väiksem väärtus tekitab tähekujulised rakud (väärtus **0.5** annab funktsiooni Minkovsky 1/2) ja suuremad väärtused ruudukujulised rakud (väärtus **4.0** annab tulemuseks funktsiooni Minkovsky 4, väärtus **10.0** funktsiooni Chebychev). Seega on enamik vahemaa algoritme matemaatiliselt samad ehk Minkowsky kauguse variandid.

Irregulaarse kujuga ümaraid rakke on võimalik saada valikutega Actual Distance (tegelik vahemaa) ja Distance Squared (vahemaa ruut).

Feature Weights (omaduste kaalud)

Need neli välja Voronoi paneeli allosas määravad neli Worley konstanti, mida kasutatakse antud kaugusalgoritmi järgi rakkudevahelise kauguse arvutamiseks. Nende väärtuste muutmine võib lõpptulemusel tekitada päris huvitavaid efekte.

Coloring (toon)

Neli välja (Intensity (intensiivsus), Position (asend), Position and Outline (asend ja piirjoon) ja Position, Outline, and Intensity (asend, piirjoon ja intensiivsus)) võimaldavad kasutada nelja erinevat mürabaasi tekstuuri värvi arvutamiseks. "Worley parameetrite" muutmine annab Voronoi tekstuurile täiesti erineva väljanägemisega ja on analoogiline mürabaasi valiku muutmise teistele tekstuuridel.

Tehnilised detailid

Worley algoritmi põhjaliku kirjeldust vaata veebilehelt: [Worley selgitus](http://amrc.altervista.org) (mittetöötav link).

Pilditekstuurid

Terminiga pilditekstuur (Image Texture) tähistatakse lihtsalt seda, et pilti — värvi (R, G, B) ja mõnikord ka läbipaistvuse (Alpha) väärtuste ristkülikukujulist pikslimaatriksit — kasutatakse tekstuuri allikana. Nagu ka teisi tekstuuritüüpe, saab pildiinformatsiooni kasutada mitmel viisil, mitte ainult "pealetrükkina".

Kui valida tekstuuritüübiks Image or Movie (pilt või film), ilmuvad kolm uut paneeli, mis annavad võimaluse häälestada suuremat osa pilditekstuuri kasutamise võimalustest: Image (pilt), Image Sampling (pildi lugemine) ja Image Mapping (pildi seostamine).

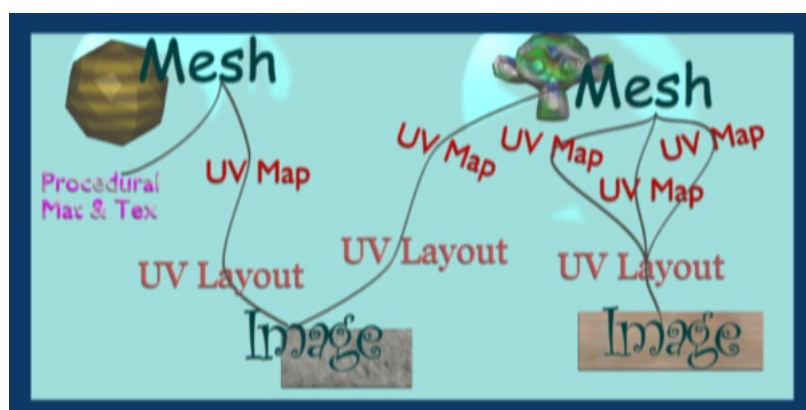
Piltidega tektuuridest

Tekstuuripildid kulutavad väärtuslikku mäluruumi. Sageli laaditakse nad ka spetsiaalsesse videomällu, mis on väga kiire ja kallis ning seetõttu tihti väikesemahuline. Seega kasuta nii väikeseid pilte kui võimalik. 64×64 pilt vajab neli korda vähem mälu kui 128×128 pilt.

Animeeritud objektide fotorealistlikul renderdamisel kasutatakse tihti suuremaid pilditekstuure, sest kaamera liikumisel võib objekt sattuda suurde plaani. Üldreeglinas tasub kasutada tekstuure, mille suurus on proportsionaalne pikslite arvuga, mis ta katab lõpprenderdusel. Lõppude lõpuks on sinu arvutis ainult piiratud hulk operatiivmälu, mida peab renderdamisel jagama pilditekstuuri, mudeli ja töökeskkonna vahel.

Et maksimeerida mälukasutuse efektiivsust, peaksid pilditekstuurid olema ruudukujulised, küljepikkusega, mis võrdub 2 astmega — nagu 32×32, 64×64, 128×128, 256×256, 1024×1024, 2048×2048 ja 4096×4096.

Kui sul õnnestub kasutada sama tekstuuri mitmel eri võrel, hoiab ka see oluliselt kokku vajaminevat mälu. Samu pilte saab kasutada paljudel võreel, kui laotada "samase väljanägemisega" võrede osad ühele pildile. Allpool toodud ülevaates kasutatakse vasakpoolset pilti nii kera kui ka osaliselt ahvi pea tekstuurina. Ahvi pea kasutab kahte laotust: ühte, mille üks UV-kaart koosneb vaid mõnest küljest, ja teist, mis koosneb kolmest UV-kaardist.



Kuidas kõik UV-tekstuurimise osad koos töötavad

Kui sa kasutad tekstuure pildifailidest, on väga oluline, et võre UV-kaart oleks [lahti lõigatud](#) ja sobivalt laotatud.

Alati ei ole vaja koostada UV-kaarti *kogu* võrele. Ülaltoodud keral on mõned küljed laotatud UV-tekstuurile, kuid ülejäänud kasutavad protseduurilisi materjale ja tekstuure. Kasuta UV-tekstuure ainult nende võre osade jaoks, mis nõuavad väga peeneid ja täpseid detaile. Näiteks vaasi mudelil vajab UV-tekstuuri ainult kaunistustega serv. Diivanipadja tagaküljel ei pea olema teistsugune pilt kui esiküljel. Tegelikult on sellise padja tagaküljel tihti hoopis kangas (protseduuriline materjal).

Veel üks näide: laota pea mudelil mõlemad silmad samale pildile (välja arvatud juhul, kui sa tahad, et üks silm oleks verd täis valgunud ja teine selge). Samas mõlema näo poole laotamine samale pildialale ei ole soovitatav, sest sünnimärgid ja nahadefektid ei ole sümmeetrilised. Sa võid muidugi muuta ühe näo poole UV-kaarti, et defekte pisut nihutada, kuid see ei pruugi anda rahuldavat tulemust. Kõrvad on samuti näide sellest, kuidas samad pildid või pildi osad võib seada vastavusse samaste võre külgedega.

Töö käik

Tegevus koosneb järgmistest sammudest.

1. Modelleeri võre. [Lõika see lahti \(Unwrap\)](#) üheks või mitmeks [UV-laotuseks \(UV Layout\)](#).
2. Loo võre jaoks üks või mitu materjali.
3. Loo iga UV-laotuse jaoks üks või mitu pilti vastavalt vajaminevatele tekstuuritüüpidele. Seejärel kas
 - maali tekstuurimaalija (*Texture Paint*) abil 3D-aknas otse võrele,
 - lae ja/või muuda pilti UV-redaktori aknas või
 - eelrenderda olemasolevad materjalid UV-redaktori jaoks pildile.
4. Määra need pildid võre UV-tekstuurideks, muutes ühte või mitut pinnaomadust. Seda saab teha, kasutades ühte või mitut paljudest väljundi seostamisviisidest (*Map To*). Näiteks,
 - seostades pildi värviga (*Color*), saad muuta võre pinna värvi,
 - seostades pildi normaaliga (*Normal*), saad muuta normaalide suunda, andes pinnale konarliku või kortsus väljanägemise,
 - seostades pildi läikiega (*Specularity*), saad muuta pinna osad läikivaks (õlitatuks).
5. Et saavutada realistlikku tulemust, kasuta üksteise peal mitut tekstuuri.

Piltide ja materjalide kasutamine

Et kasutada pilti tekstuuri värvi ja läbipaistvuse (*alpha*) jaoks, võid sa luua pildi välises programmis ja avada vastava faili UV/pildiredaktoris kui tekstuuri. Või siis luua redaktoris uue pildi ja salvestada selle tekstuurina.

Kui sa tahad joonistada pildi välises programmis, siis sa tõenäoliselt tahad kõigepealt salvestada oma UV-laotuse joonise, kasutades menüü *UVs* valikut *Save UV Face Layout* (salvesta UV-külgede laotus). UV-laotuse salvestamist kirjeldatakse [selles](#) peatükis.

Pilditekstuuri loomine

Blenderis pildi loomiseks pead sa kõigepealt [looma uue](#) ühevärvilise või testruudustikuga pildi. Pärast seda saad sa seda pilti muuta kas:

- Kasutades tippude värve pildi värvide määramiseks või
- Eelrenderdades pildile võre tegeliku väljanägemise stseenis

Kui pilt on loodud, saad sa seda muuta Blenderi sisseehitatud [tekstuuriimaali](#) (*Texture Paint*) või välise pilditöötlusprogrammi abil.

Sa näed tekstuuri 3D-aknas, aga mitte renderdamisel

Kui esitusviis on "tekstuuritud", on kasutatav tekstuur näha 3D-aknas. Sellest piisab ka selleks, et tekstuur oleks näha Blenderi mängumootoris. Renderdamiseks on aga objektile vaja lisaks tekstuurile ka materjali. Kui sa tahad võre renderdada UV-tekstuuridega, pead talle lisama küljetekstuuride (Face textures) materjali. Materjalipaneelil vali Add New Material (lisa uus materjal) ja lülita valik Face Textures (küljetekstuurid) sisse.

Näited

Tegelase näo jaoks võib olla üks UV-laotus ja riiete jaoks teine. Riiete tekstuurimisel pead sa looma pildi vähemalt riiete värvuse määramiseks. Võimalik, et vaja läheb ka täiendavat normaalipilti (faktuuripilti), andmaks pinnale faktuurse kootud kanga väljanägemise. Kui kangas katab näiteks ka õlgu ja põlvi, on kanga läige neis kohtades erinev ja sa vajad lisapilti, mis määrab Blenderis kanga läike (*Specularity*) igas võre piirkonnas. Kui kangas on voltis või kortsus, tuleb sul lisada veel pilt, mis määrab pinna nihke võrel (deformeerib pinda füüsiliselt). Kõik need on näited võrele tekstuuri lisamisest.

Teine hea näide on inimnägu, mille kohta on alati palju küsimusi ja ka õppetükke. Alustuseks peaksid sa looma materjali, mis määrab naha põhivärvi, vajalikud varjutusalgoritmid ja pinnaaluse hajumise (*subsurface scattering*). Seejärel tuleb sul lisada sellele täiendavad UV-tekstuurid:

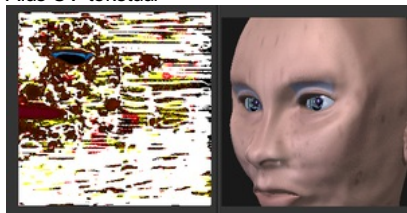
- Sünnimärkide pilt - värvid ja normaaliid
- Nahaalused veenid ja kõõlused - nihe
- Kortsud, vaod ja nahapaksendid - normaaliid
- Meik - värvid
- Naha läige - läige
- Zombi jaoks läbipaistvus neile kohtadele, kust liha on ära mädanenud (*vuih!*)
- Varjutus lõua all ja ninasõõrmetes
- Õhuke nahk on läbipaistvam, seega on ka selle jaoks vaja eraldi kaarti

Iga järgmine pilt lisatakse uude tekstuurikanalisse (*Texture Channel*). Kõik need tekstuurid on pildid, mis määravad lõpliku pildi erinevaid aspekte (värv, normaaliid, läige). Korduvaid (*tileable*) pilte saab tekstuuri jätku ja suuruse seadistuste abil panna võrel korduma, moodustades tiheda kompaktse mustri.

UV-tekstuuride üksteise peale panemine



Alus UV-tekstuur



Kattev UV-tekstuur

Asetades pilte üksteise peale, saab luua suurepäraseid tekstuure. Alusta baaskihist, mis on objekti põhivärv. Iga järgnev kiht on (osaliselt) läbipaistev nendes kohtades, kus aluskiht peab paistma, kuid läbipaistmatu seal, kus sa tahad detaile lisada.

Et vältida totaalset segadust, on kõigil võre pilditekstuuriid enamasti sama UV-laotus. Kui sa seda reeglit järgid, sobivad osapildid täpselt üksteise peale ja moodustavad ülekattes lõpliku tekstuuri, nagu parempoolses näites. Selleks koosta vastava peatüki juhtiste järgi üks UV-tekstuur (tekstuurikaart). Seejärel lisa materjalile pilditekstuurid, nagu kirjeldatakse protseduuriliste materjalide lõigus. Automaatsete koordinaatide (*OrCo*) asemel seosta pildikoordinaadid UV-kanaliga (*UV*).

Kasuta sama UV-kaarti kõigi tekstuuride jaoks (sea tekstuurikonteksti paneelil *Mapping* (koordinaatide seostamine) välja *Coordinates* (koordinaadid) väärtuseks UV ja sisesta väljale *Layer* (kiht) sama nimi). Parempoolsel pildil (NB! Blender 2.4 paneel) on UV-tekstuurikaardi nimi "Head" (pea) (et paneeli näha, pead sa võib-olla pilti suurendama). Selle tulemusena näidatakse objektile pilditekstuuri vastava UV-laotuse koordinaatide järgi. Parempoolsel näitel toodud näol on kaks pilditekstuuri: üks naha baasvärv ja teine täppide, iluvigade ja meigi jaoks.

Mõlemad tekstuurid kasutavad sama UV-kaarti ja mõlemate väljund on seostatud materjali värvusega. Meigitekstuur on läbipaistev, v.a värvilistes kohtades, ja seega paistab nahapind temast läbi. Pane tähele, et selle pildi värvid olid liiga tugevad ja nii on teise (meigi) pildi värvuse määra alandatud 60%-ni.

Tavaliselt me eeldame, et pilditekstuurid muudavad võre (materjali) värvi. Realism ja fotorealistlik renderdamine on kombinatsioon paljudest erinevatest viisidest, kuidas pind mõjutab talle langetavat valgust. Pilditekstuuri väliundit saab seostada mitte ainult värvi, vaid

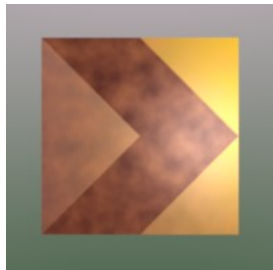
ka normaalidega (Normals), andes pinnale faktuuruse, või peegeldusega (Reflection) või ükskõik millise muu valikuga materjaliga seostamise paneelil *Map To*.

Kui sa joonistad halltoonides pildi (mis on paigutatud vastavalt UV-laotusele), millel heledad toonid märgivad kohti, kus nahk on õline ja läikiv, ja tumedad, kus nahk on matt, siis kasutad sa küll sama UV-kaarti, kuid seostad selle tekstuuril väljundi materjali läike, mitte värviga.

Et muuta osa võrest läbipaistvaks nii, et oleks näha teine võre pind esimese all, pead sa maalima halltoonides pildi, kus must märgib kohti, kus tekstuur peab läbi paistma. Siis kasutad koordinaatidena UV-kaarti ja seostad tekstuuril väljundi mitte materjali värvi, vaid läbipaistvusega (*alpha*). Et panna osa võrest helendama (näiteks hõõguva metalli kujutamiseks), kasutad sa halltoonides pilti ja seostad väljundi materjali kiirgusega (*Emit*).

Usu või ära usu, aga see on ainult "jäämäe veepealne osa!" Kui kõigest, mis siin on kirjeldatud, sulle ei piisa, võid kasutada uue Blenderi versiooni *tekstuurisõlmi*, mis võimaldavad sul kombineerida tekstuure praktiliselt ükskõik millistel kujutletavatel viisidel.

Materjalide segamine ja ühendamine



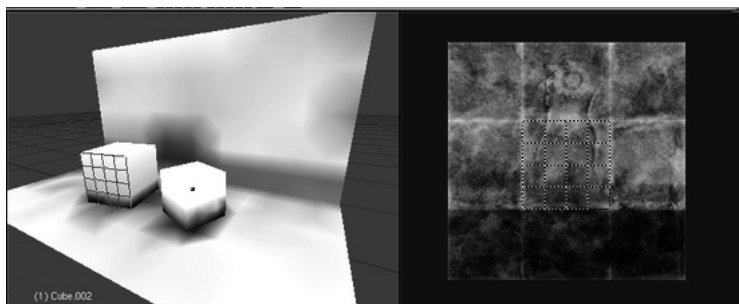
Sa saad samal võrel segada protseduurilisi materjale ja tekstuure, tipuvärve ning UV-tekstuure.

Parempoolse pildi stseenis on punane hajusvalgus. Materjalil on nii tipuvärvid (*Vertex Colors Paint*) kui ka küljetekstuudid (*Face Textures*) sisse lülitatud ja teda valgustab poole intensiivsusega hajusvalgus. Nõrk pilvetekstuur muudab värvust, lisades sellesse pronksjat tooni. Parempoolsed tipud on värvitud kollaseks ja vasakpoolsed on jäetud värvimata (protseduuriline hall). UV-tekstuur on standardne noole pilt vabalt kasutatavate piltide CD-lt. Stseeni valgustab valge lamp paremal pool. Antud informatsiooni ja eelneva kasutusjuhendi materjalide järgi peaks sa suutma selle stseeni taasluua.

Sa saad võrele lisada ka [mitu materjali](#) vastavalt sellele, millised küljed tahad sa jätta protseduurilisteks ja millised katta tekstuuriga. Lihtsalt ära tee UV-laotust nendele külgedele, mis peaksid jääma protseduurilisteks.

Kui materjalil on UV-tekstuudid ja tipuvärvid (*Vertex Paint*) (kiirklahv V 3D-aknas) sisse lülitatud, saab neid mõlemaid korruga kasutada. Tipuvärve saab kasutada kas UV-pilditekstuuri heleduse või värvi muutmiseks:

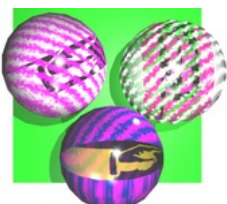
- Põhjaks on UV-tekstuur (küljetekstuudid (*Face Textures*))
- Tipuvärvid muudavad tekstuuril värvi, seejärel
- Protseduurilised tekstuudid asetatakse nende peale,
- Koht- ja suunatud valgus valgustab pinda, heidab varje jne ja lõpuks
- Üldvalgus valgustab kogu stseeni.



Tipuvärvid muudavad tekstuuril.

Ühel UV-laotusel saab olla ainult üks pilt, ehkki seda pilti saab piki pinda korrata ja animeerida. Kuna laotus on kogum sobivalt paigutatud UV-kaarte ja UV-kaardid määravad võre külgede laotused, saab seega ühel võre küljel olla ainult üks UV-pilt ja vastava külje UV-koordinaadid peavad täielikult mahutama selle pildi sisse. Kui sa tahad ühel võre küljel kasutada mitut pilti, lõika see külj osadeks ja määra igale osale erinev pilt. (*Lisaks* saad luua erilisi efekte materjalisõlmedega, aga see on eraldi jutt...)

Läbipaistvuse (*Alpha Transparency*) kasutamine



Läbipaistvad UV-tekstuudid

UV-pildi alad, mille läbipaistvuse (*alpha*) väärtus on 0.0 (läbipaistvad), renderdatakse mustana. Erinevalt protseduurilistest tekstuuridest ei muuda nad alusmaterjali läbipaistvaks, sest UV-tekstuudid ei töötle protseduurilist alusmaterjali. UV-tekstuur kirjutab allpool oleva protseduurilise värvi väärtuse üle. Protseduurilised tekstuudid pannakse UV-tekstuuride peale, seega kirjutavad

protseduurilised pilditekstuurid UV-tekstuuride pildi üle. Läbipaistvad (mustad) protseduuriliste tekstuuride alad, mis on seostatud läbipaistvusega, mõjuvad kõige viimasena ja muudavad objekti nendes kohtades läbipaistvaks. Ainus asi, mis muudab nähtavat UV-tekstuuri, on tipuvärv. Parempoolsel näitel on UV-tekstuurina kasutatud sõrme pilti, mis on läbipaistev randme kohalt ja sõrme ülaosast. Kõigi kolme kuuli baasmaterjal on sinist värvi ja protseduurilise marmoritekstuuriga. Kui küljetekstuurid (*Face Textures*) on sisse lülitatud, siis baasmaterjali värvi ei kasutata.

Ülemisel vasakpoolsel kuulil ei ole tipuvärve kasutatud, sõrm on paigutatud kuuli ekvaatorile ja tekstuuri väljund on seostatud roosa värviga. Nagu näha, on baasmaterjalil tippude värvid (*Vertex Color Paint*) ja küljetekstuurid (*Face Textures*) sisse lülitatud; sinist alusvärvi ei kasutata, küll aga tekstuuri. Kui tipuvärve ei ole, ei muuda miski UV-tekstuuri värve ja seega paistab sõrm valge. UV-pildi läbipaistvad alad on mustad.

Ülemisel parempoolsel kuulil on vertikaalsel külgede ribal roosa tipuvärv (3D-aknas vali UV-maalimise (*UV Paint*) režiimis küljed, lülitu tippude maalimise (*Vertex Paint*) režiimi, vali roosa värv ja seejärel menüüst Paint->Set Vertex Colors (maali->määra tippude värv)). Sõrm on paigutatud keskmisele meridiaanile ja tippude värv (*Vertex Color Paint*) ning küljetekstuurid (*Face Textures*) on sisse lülitatud. Protseduuriline tekstuur on seostatud läbipaistvusega (*alpha*) ning tema väärtus korrutatakse alusmaterjali läbipaistvusega, milleks on 1.0. Tekstuuri valgete alade läbipaistvus on 1.0 ja kuna 1.0 korda 1.0 on 1.0, siis on need alad läbipaistmatud ning nähtavad. Protseduurilise tekstuuri mustade alade väärtus on 0.0 ja korrutamise tulemusena muutub materjal nendes kohtades läbipaistvaks. Nagu näha, määrab ilma UV-laotusega alade värvi (kuuli vasak ja parem pool) üksnes tipuvärv (kui see puudub, siis hall) ja värvitud alad on roosad. Keskmisel alal, mis on ühtaegu värvitud ja millel on UV-laotus, muudab tipuvärv UV-tekstuuri valged alad roosaks. Kohtades, kus protseduuriline tekstuur teeb objekti läbipaistvaks, on näha roheline taust. UV-tekstuuri läbipaistvad alad on ikkagi mustad.

Alumine pall kasutab mitut materjali. Suurem osa kuulist (kõik peale keskmise riba) on alusmaterjal, millel ei ole küljetekstuuri (*Face texture*) ega ka tippude värve (*Vertex Color Paint*) sisse lülitatud. Ilma nendeta paistab välja sinine alusmaterjali värv, mida moduleerib talle lisatud roosa protseduuriline tekstuur. Keskmisele ribale on lisatud uus materjal (2 Mat 2), millel on tipuvärvid (*Vertex Color Paint*) ja küljetekstuurid (*Face Textures*) sisse lülitatud. Keskmise riba tipud on värvitud kollaseks, seega paistavad sõrme valged alad kollastena. Nendes kohtades, kus roosa protseduuriline tekstuur katab UV-tekstuuri, on summaarne värv roheline (sest roosa ja kollane annavad kokku rohelist).

Kui sa tahad, et üks pilt paistaks läbi teise ja seguneks temaga, tuleb kasutada läbipaistvust (*alpha*). Alusmaterjalil võib olla pilditekstuurile määratud läbipaistvus (*alpha*), mis laseb UV-tekstuuril sellest läbi paista.

Mitme UV-tekstuuri üksteise peale asetamiseks on mitu võimalust:

- Loo mitu UV-tekstuuri sama UV-kaardiga ja kasuta igaühe jaoks neist erinevat pilti (läbipaistvate aladega) ning Blender paigutab nad õigesti üksteise peale.
- Kasuta [komposiitmissõlmi](#) ja kombineeri kaks pilti läbipaistvussõlmega AlphaOver, luues uue ühendatud pildi ja salvestades selle. Ava see ühendatud pilt ja kasuta UV-tekstuuriks.
- Kasuta välist pilditötlusprogrammi ja aseta seal kaks poolläbipaistvat pilti üksteise peale, salvesta tulemus ja laadi see UV-tekstuuriks.
- Loo kaks objekti, üks veidi teise sees. Sisemisel objektil on alustekstuur ja välisel objektil pealmine tekstuur, kusjuures tema materjali läbipaistvus (*alpha*) olgu väiksem kui 1.0.
- Kombineeri need kaks pilti [materjalisõmede](#) abil, kasutades läbipaistvussõlme AlphaOver või segamissõlme Mix uueks materjaliks ning määra võre vastavatele külgedele see materjal. Selle meetodi puhul ei vaja sa UV-kaarti, vaid võid lihtsalt lisada selle materjal küljele, kasutades mitme materjali (*Multiple Materials*) valikut.

UV-tekstuurid vs protseduurilised tekstuurid

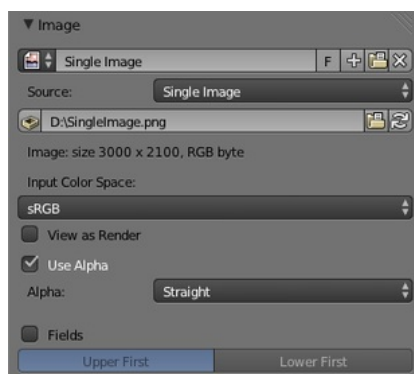
Materjalitekstuur, mille koordinaatseos (*Map Input*) on UV, mis on pilditekstuur ja mille väljund on seostatud (*Map To*) värviga, on samaväärne UV-tekstuuriga. Selline tekstuur on palju paindlikum, sest tema suurus, nihet ja määra, kui palju ta muudab objekti värvi, saab muuta väljundi seostamise paneelilt *Map To*. Lisaks võid sa iga tekstuurikanali jaoks kasutada erinevat pilti: ühte värvi, ühte läbipaistvuse, ühte normaali, ühte läike, ühte peegelduse jaoks jne. Protseduurilised tekstuurid, nt pilved, on ERAKORDELT lihtsad ja kasulikud stseenile realismi lisamiseks.

UV-tekstuur	Protseduuriline tekstuur
Pilt laotatakse täpsete koordinaatide järgi valitud võrekülgedele	Muster genereeritakse dünaamiliselt ja laotatakse kogu võrele (või osale, mis on kaetud selle materjaliga)
Pilt laotatakse korraga eelnevalt valitud võrekülgedele	Laotatakse korraga külgedele, millel on vastav materjal, kas tervele võrele või selle osale
Pilt laotatakse külgedele ühekordselt.	XYZ Size (suurus) sisend-seosel <i>MapInput</i> võimaldab tekstuuri piki võre külgi korrata. Korduste arv sõltub võre suurusest
Muudab objekti värvi ja läbipaistvust.	Võib muuta ka normaale (faktuuri), peegeldust, helendust, nihet ja paljusid teisi võre väljanägemise aspekte. Võib isegi moonutada ja varjata järgnevaid tekstuure.
Võrel võib olla mitu tekstuuri	Tekstuure saab asetada üksteise peale (kuni 10 tekstuuri). Palju segamismeetodeid eri kanalite omavaheliseks segamiseks.
Suvaline pilditüüp (üksikpilt, video, renderdatud pilt). Saab kasutada genereeritud testvõret.	Valik hulga eeldefineeritud variantide seast: pilved, puusüü, marmor, müra ja isegi maagia.
UV-laotus töötab ka animeeritud tekstuuride puhul	Ainuke animeeritud tekstuur on müra
Kulutab väärtuslikku videomälu	Kasutab väga vähe mälu, selle asemel kulutab protsessoriaega

Kokkuvõtteks võib öelda, et ühtainust UV-tekstuuri on võrel küll lihtsam kasutada, aga tema võimaluste hulk on piiratum, võrreldes mitme (UV-kaardi järgi laotatud) tekstuuriga. Viimased teevad väga hästi ühte asja: lisavad võre vastavatele külgedele detaile. Neid saab ka koos kasutada, kui protseduuriliste tekstuuride koordinaadid siduda UV-laotusega. Nagu eelpool öeldud, saad sa sisendi seostamise paneelil *Map Input* UV-koordinaatide sidumise abil kasutada erinevatel UV-tekstuuridel erinevaid pilte.

Sätted

Pilt (*Image*)



Pildipaneel Image

Pildipaneelis Image saab määrata, milist pildifaili Blender kasutab.

Pilditekstuur (Image texture)

Sirvija (Browse)

Valib pildi dokumendis kasutatavate piltide hulgast

Nime väli (Name)

Pildi nimi selle dokumendi piires

F

Loo sellele pilditekstuurile võltskasutaja

+

Asendab aktiivse tekstuuri uuega

Kaust (Folder)

Võimaldab sirvida (ja avada) arvutis olevaid pilte

X

Ühendab selle pildi lahti (eemaldab tekstuurilt)

Allikas (Source)

Määrab kasutatava failitüübi.

Fail (File)

Paki pilt (Pack image)

Pakib pildi käesoleva .blend-faili sisse

Asukoht (Paht)

Faili asukoht

Failisirvija (File Browser)

Võimaldab sirvida (ja avada) arvutis olevaid pilte. Hoia all ⇧ Shift, et avada valitud fail ja Ctrl, kui tahad sirvida kataloogi, kus see fail asub.

Laadi uuesti (Reload)

Loeb pildi failist uuesti. See on kasulik siis, kui pilti on välises programmis muudetud.

Poolkaadrid (Fields)

Kasutatakse poolkaadritest koosnevate piltide puhul. Videokaadrid koosnevad tavaliselt kahest järjestikusest nihutatud pildist (poolkaadrist), mis liidetakse kokku. See valiku abil saad tagada, et poolkaadritega video renderdamisel kasutatakse õiges kohas õiget poolkaadrit. MIP Mapping koos poolkaadritega ei tööta. Tavaliselt algab ülerealaotusega video esimene poolkaader esimeselt realt (Upper First, ülemine enne). Mõned videodigitaliseerijad kasutavad aga vastupidist järjekorda (Lower First, alumine enne).

Eelkorrupta (Premultiply)

Eelkorruptamine tähendab, et pildi RGB kanalite väärtused korrutatakse enne pildi võrele kandmist läbipaistvuse (*alpha*) väärtusega.

Pildijada/film (Sequence/Movie)

Kaadreid (Frames)

Kaadrite arv, mida filmist või järjestusest kasutada

Algus (Start)

Jada/filmi alguskaader Blenderis

Nihe (Offset)

Kaader filmis/jadas, millest alustatakse animatsiooni

Poolkaadrid (Fields)

Film kasutab poolkaadreid. Määrab, kas video on ülerealaotusega (iga kaader koosneb kahest poolkaadrist) või mitte.

Uuenda automaatselt (Auto Refresh)

Loeb kaadri vahetusel pildi uuesti sisse

Korduv (Cyclic)

Pärast video lõppemist alustab mängimist uuesti algusest.

Genereeritud (Generated)

Suurus (Size)

Genereeritud pildi laius ja kõrgus

Tühi/UV-võrk/värvivõrk (Blank/UV Grid/Color Grid)

Milline pilt genereerida

Pildi lugemine (*Image sampling*)

Pildilugemise paneelil Image Sampling saab määrata, kuidas ja millist informatsiooni pildist loetakse.

Nende kahe pildi abil näitame erinevate pildilugemisvalikute mõju. *Taustapilt* on tavaline JPG fail, *esiplaani pilt* on PNG-fail varieeruva läbipaistvusega (*alpha*) ja mitmesuguste halltoonide väärtustega. Esiplaani pildi kõige parempoolsem tulp on läbipaistev värvüleminek. Horisontaalse rea läbipaistvus on 50%.



Vasak: taustapilt
Parem: esiplaani pilt

Alpha (läbipaistvus)

Pildi läbipaistvusega seotud valikud

Use (kasuta)

Töötab PNG ja TGA pildiformaatidega, mis sisaldavad läbipaistvusinformatsiooni (vaata kõrvalolevat pilti). Kui läbipaistvuse (*alpha*) väärtus pildil on väiksem kui 1.0, on see koht osaliselt läbipaistev ja tagaplaan paistab.

Calculate (arvuta)

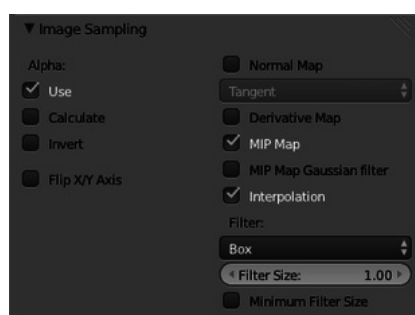
Arvutab läbipaistvuse väärtuse pildi RGB väärtuste järgi. Must (0,0,0) on läbipaistev, valge (1,1,1) läbipaistmatu. Lülita see valik sisse siis, kui pilditekstuur on mask. Pane tähele, et halltoonid maskis annavad poolläbipaistva tekstuuri, mis sobib nt vaimudele, leekidele ja suitsule/udule.

Invert (pööra ümber)

Pöörab läbipaistvuse väärtuse ümber. Kasuta seda valikut siis, kui maskis on valgega märgitud läbipaistvad ja mustaga läbipaistmatud alad.



Vasak: esiplaani pilt siis, kui Use Alpha (kasuta läbipaistvust) on sisse lülitatud. Pikslite läbipaistvus saadakse hinnangulisena
Parem: esiplaani pilt siis, kui Calculate Alpha (arvuta läbipaistvus) on sisse lülitatud



Pildi lugemise paneel Image Sampling

Normal Map (normaalikaart)

Annab Blenderile teada, et seda pilti tuleb tõlgendada normaalikaardina faktuurse pinna loomiseks ja selle RGB kanalid määravad pinna näiliste ebatasasuste varjud. See meetod vajab erilist koostatud sisendpilte. Vaata peatükki [faktuuri- ja normaalikaardid](#).

Normal Map Space (normaalikaardi koordinaadistik)

Tangent (tangentsiaal)
Object (objekti)
World (globaalne)
Camera (kaamera)

MIP Map

[MIP Mapid](#) on automaatselt eeltöödeldud, pisemad, filtreeritud kindla suurusega tekstuurid. Genereeritakse seeria pilte, millest iga järgmine on eelmisest poole väiksem. See optimeerib tekstuuride filtreerimist. Vaikimisi on see valik sisse lülitatud ja muudab renderduse kiiremaks (eriti kasulik mängumootoris). Kui see valik on välja lülitatud, on tekstuurid tavaliselt teravamad, kuid arutusaeg filtri mõõtmete suurenemisel (vaata allpool) muutub pikaks. Ilma MIP Mape kasutamata võib tekstuur väikesel suurendusel iga kaamera liikumise puhul virvendada. Animatsioonide puhul on see märgatav.

MIP Map Gaussian filter (MIP Mapi Gaussi filter)

Kui MIP Map on sisse lülitatud, võimaldab see MIP Mapi tekstuure värvide sarnasuse põhjal vähendada. Mängumootoris soovivad sa kindlasti, et tekstuurid (eriti MIP Mapi tekstuurid) oleksid võimalikult väikesed, sest see suurendab renderduskiirust (kaadrisagedust).

Flip X/Y Axis (vaheta X/Y-teljed)

Pöörab renderdamisel pilti 90 kraadi vastupäeva.

Interpolation (interpoleerimine)

Interpoleerib (keskmistab kaaludega) pildi piksleid. Selle mõju muutub nähtavaks pilti suurendades. Vaikimisi on see valik sisse lülitatud. Lülita see välja, kui tahad, et üksikud pikslid oleksid näha, või kui teravad piirid on juba pildil silutud (*antialias*). Seda viimast võimalust on praktiline kasutada mustrite, näiteks joonte ja ruudustike puhul, sest sel juhul jäävad nad 'teravaks' ka siis, kui pilti oluliselt suurendada. Kui sa suurendad seda 10×10-pikslit pilti, on erinevus interpolateerimise puudumise ja olemasolu vahel selgelt nähtav. Lülita see valik välja digifotode kasutamisel, et säilitada teravust.



Suurendus pilditekstuurst ilma interpolateerimiseta ja koos intepoleerimisega (Interpolation)

Filter

Filtri suurus, mida kasutatakse renderdamisel ning valikutest MipMap ja Interpolation (interpoleerimine). Kui sa märkad halle jooni või piirjooni tekstuuritud objekti ümber, eriti seal, kus pilt on läbipaistev, vähenda selle väärtust 1.0-st umbes 0.1-ni.

Texture Filter Type (tekstuuri filtri tüüp)

Filter, mida kasutatakse tekstuuripildi projitseerimisel. Nii nagu *piksel* kujutab pildielementi (*PIC*ture *ELE*ment), kujutab

"teksele" endast tekstuurielementi (*TEXTure ELEMENT*). Kui tekstuur (2D-tekstuuriruum) projitseeritakse 3D-mudelile (3D-mudeliruum), on võimalik kasutada mitmeid algoritme määramaks, kuidas iga piksli väärtus mitme teksti väärtustest leida.

Box (kast)

Kiire ja lihtne lähima naabri interpolatsioon, tuntud ka Monte Carlo integreerimisena

EWA

Elliptiline kaalutud keskmine (**Elliptical Weighted Average**) — üks kõige efektiivsemaid otsesekonvolutsiooni algoritme, mille töötasid välja Paul Heckbert ja Ned Greene 1980ndatel aastatel. EWA loeb, kaalub ja liidab iga piksli puhul kokku tekstlid ellipsikujulises alas ning jagab tulemuse kaalude summaga.

FELINE (kiired elliptilised jooned)

Fast Elliptical Lines kasutab mitut isotroopset proovi mitmes punktis piki ühte tekstuuriruumi joont. See on anisotroopne filter, mis vähendab pildisakke ilma renderdamise aega oluliselt pikendamata.

Probes (proovid)

Proovide arv, mida võtta. Täisarv 1 ja 256 vahel.

Area (pindala)

...

Eccentricity (ekstsentrilisus)

Maksimaalne ekstsentrilisus. Suured väärtused vähendavad kaugete ja suure kalde all olevate pindade udusust, kuid kulutavad rohkem aega

Algoritmi täpsem kirjeldus

McCormack, J; Farkas, K; Perry, R; Jouppi, NP (1999) [Simple and Table Feline: Fast Elliptical Lines for Anisotropic Texture Mapping](#), WRL

Eccentricity (ekstsentrilisus)

Maksimaalne ekstsentrilisus. Suured väärtused vähendavad kaugete ja suure kalde all olevate pindade udusust, kuid kulutavad rohkem aega

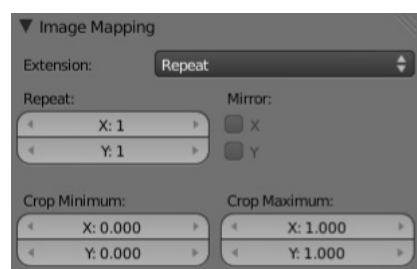
Filter Size (filtri suurus)

Filtri suurus, mida kasutatakse MIP Mapi genereerimisel ja interpoleerimisel

Minimum Filter Size (minimaalne filtri suurus)

Määrab filtri minimaalse suuruse pikslites

Pildi laotamine (*Image Mapping*)



Pildilaotuse paneel Image Mapping

Pildilaotuse paneel Image Mapping võimaldab määrata, kuidas pilt laotatakse või projitseeritakse 3D-mudelile.

Extension (laiendamine)

Extend (laienda)

Väljaspool pildi ääri jätkatakse äärte värvi

Clip (lõika)

Laotus lõpetatakse pildi äärel ja edasised pikslid on läbipaistvad. Väljaspool pilti on teksturi läbipaistvuse (*alpha*) väärtus 0.0. See võimaldab 'kleepida' suurele objektile väikese logo.

Clip Cube (lõika kuup)

Lõikab ümber pildi ära kuubikujulise ala, edasised pikslid on läbipaistvad. Sama nagu *Clip* (lõika), kuid nüüd arvutatakse ka 'Z'-koordinaat. Väljaspool pilti ümbritsevat kuubikujulist ala on teksturi läbipaistvuse (*alpha*) väärtus 0.0.

Repeat (korda)

Pilti korratakse horisontaal- ja vertikaalsuunas

Repeat (korda)

X/Y korduste arv

Mirror (peegelda)

Peegeldab pildi igal kordusel piki X/Y-telge vastava telje suunas. See valik võimaldab laotada pilti peegelduste seeriana vastavalt X- ja Y-telgede suunas.

Checker (ruudustik)

Kordab pilti malelauamustris. Laotamise paneeli Mapping valik Size (suurus) määrab malelauakorduste arvu.

Even/Odd (paaris/paaritu)

Lülitab teksturi paaris ja paaritutel ruutudel sisse ja välja

Distance (kaugus)

Määrab ruutude vahemaa suhtena teksturi suurusesse

Crop Minimum/Crop Maximum (lõike miinimum/lõike maksimum)

Textuuri alguse ja lõpu suhtelised asukohad pildil. Pikslid väljaspool seda ala ignoreeritakse. Selle abil saad lõigata (valida) suuremast pildist ala, mida kasutad tekstuurina.

Videotekstuurid

Videotekstuure saab lisada samamoodi nagu pilditekstuure. Märkus: Blender lõikab filmi läbi 100 kaadri pealt, seega pead sa kasutama valikut *Match Movie Length* (vastavalt filmi pikkusele).

Tekstuuri sõlmed

Alternatiivina [tekstuuride pinule \(Texture Stack\)](#), on Blenderis sõlmedel ja seostel põhinev tekstuuride genereerimise süsteem, mis võimaldab sul luua uusi teksture, kombineerides värve, mustreid ja teisi teksture – põhijoontes samal viisil, nagu sa kombineerid [materjalisõlmi \(Material Nodes\)](#).

Sa saad kasutada neid kõikjal, kus tavalisi teksturegi – paigutada tekstuurikanalitesse, materjalisõlmedesse, osakeste süsteemi ja isegi teiste tekstuuride sisse.

Sõlmede võrgustikes ("nuudlites") on kolme põhilist tüüpi sõlmi: **sisendsõlmed**, **filtersõlmed** (ehk **muutmissõlmed**) ja **väljundsõlmed**. Sa võid võrgustikku paigutada suvalise arvu neid sõlmi ja ühendada neid lugematul arvul viisidel. See annab sulle piiramatult loominguilise ja tehnilise vabaduse.

Märkus

Sõlmetekstuurid **ei tööta** reaajas (3D-aknas), vaid nad on näha ainult renderdatud piltidel.

Tekstuurisõlmede kasutamine

Et kasutada tekstuurisõlmi aktiivsel tekstuuril, ava [sõlmeredaktor](#) ja sea see tekstuurirežiimi Texture Mode, vajutades "tekstuuri" ikoonile päises.

Et sõlmi lisada, pead sa kõigepealt valima materjali. Nüüd võid sa vajutada sõlmeredaktoris nuppu New (uus) või nuppu New (uus) tekstuuripaneelil. Kui tekstuur on valitud, võid valida, kas see töötab tavalise- või sõlmetekstuurina, vajutades sõlmeredaktoris nupule Use Nodes (kasuta sõlmi).

Ilmub vaikimisi sõlmede paigutus: puna-valge malelauasõlm, mis on ühendatud väljundsõlmega Output, mille nimi on "Default". Sa võid oma sõlmede võrgustikus lisada *tekstuurisõlmedele* nii palju väljundeid kui tahad. (Teised sõlme võrgustiku tüübid, nagu sa võib-olla mäletad, lubavad ainult ühte väljundsõlme.) Vaata täpsemalt järgmisest peatükist.

Juhised selle kohta, kuidas lisada, eemaldada ja muuta sõlmi, leiad [sõlmeredaktori juhendist](#).

Mitme väljundi kasutamine

Igal tekstuurisõlmede abil loodud tekstuuril võib olla mitu väljundit, millest igatühte saab kasutada eri asjade jaoks. Näiteks võid sa soovida, et su tekstuur määraks nii alusvärvi (*diffuse color*) kaardi kui ka normaalikaardi. Et seda teha, pead sa:

1. looma tekstuuride nimekirja kaks tekstuurivälja ja siduma nad mõlemad sama tekstuuriblokiga (*texture datablock*).
2. lisama sõlmede puule kaks väljundsõlme Output ja sisestama kummagi tekstiväljale Name (nimi) unikaalsed nimed: *näiteks* "Alusvarv" ühe jaoks ja "Normaal" teise jaoks.
3. Tekstuuripaneelil, tekstuurivalija all, näed sa menüüd sinu väljundsõlmede nimedega. Vali tekstuuride nimekirjas iga välja jaoks sellest menüüst soovitud väljundsõlm (st seo üks sõlmega "Alusvarv" ja teine sõlmega "Normaal").

Neid samu (nimelisi) väljundeid saad kasutada ka sel juhul, kui lood materjali materjalisõlmede abil. Sel juhul sa tõenäoliselt ei kasuta küll tekstuurikanaleid. Selle asemel lisad sa tekstuurisõlmed materjali sõlmede puusse, valides Add (lisa) → Input (sisend) → Texture (tekstuur). Siis saad sa vastset lisatud tekstuurisõlme sees määrata, millist väljundit sa tahad kasutada (*st kas* Alusvarv või Normaal).

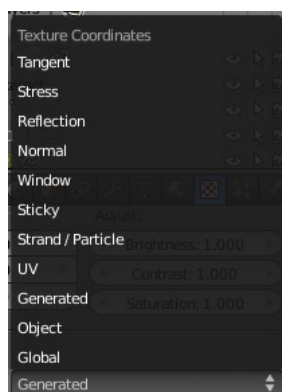
Vaata ka

- [Arenduslehekülge](#)

Tekstuuri laotus

Et määrata, kuidas tekstuurid objektile paigutada, on neile vaja anda laotuskoordinaadid. Laotus määrab, kuidas tekstuur on ümber võre "mähitud". Näiteks saab 2D-pilditekstuuri seada rulli ümber silindrikujulise objekti.

Koordinaadid



Laotuskoordinaadistiku menüü Coordinates

Koordinaatide laotamiseks on vaja koordinaate, mis määravad tekstuuri paigutuse. Koordinaatide allikas võib olla suvaline, kuid tavaliselt loetakse nad samalt objektilt, millele tekstuur laotatakse.

Global (globaalne)

Stseeni globaalne 3D-koordinaatide süsteem. See on kasulik animatsioonide puhul: kui sa liigutad objekti, liigub tekstuur piki objekti pinda. Nii saab lasta objekti teatavas ruumpunktis tekkida või kaduda.

Object (objekt)

Loe tekstuurikoordinaadid teiselt objektilt. Tihti kasutatakse seda koos tühiobjektiga Empty, nii saab lihtsalt paigutada väikese pildi objekti peale (vaata [näidet allpool](#)). See objekt võib olla animeeritud; sel juhul liigub tekstuur piki pinda või läbi selle.

Object (objekt)

Vali teise objekti nimi.

Generated (genereeritud)

Objekti algsed, deformeerimata koordinaadid. See on tekstuuride laotamise vaikevalik.

UV

UV-laotus on väga täpne meetod 2D-tekstuuri laotamiseks 3D-pinnale. Igal võre tipul on oma UV-koordinaadid. Võre lõigatakse lahti ja laotatakse tasapinnaliselt nagu nahk. Sa võid mõelda UV-koordinaatidest kui üksühesest vastavusest 2D-pinna (koos oma kohaliku koordinaatsüsteemiga) ja võre pinna vahel. See laotus on eriti praktiline siis, kui tekstuuridena kasutada 2D-pilte, nagu kirjeldatud [UV-laotuse](#) peatükis. Ühe UV-koordinaatide komplektiga võib seostada mitu tekstuuri.

Layer (kiht)

Vali UV-kiht, mis määrab laotuse.

Strand/Particle (kiud/osake)

Kasuta kiu normaliseeritud 1D-koordinaati või osakese iga (*age*, *X*) ja asendit trajektoril (*position*, *Y*). Kasuta seda siis, kui tekstuur kantakse juuksekarvadele või osakestele.

Sticky (kleepuv)

Kasutab võre kleepuvaid koordinaate ehk tipupõhiseid UV-koordinaate. Kui sa oled eelnevalt loonud kleepuvad koordinaadid (tavaliselt kaamera vaade → Space → kirjuta sticky → vali Add Sticky/Remove Sticky (lisa kleepuv/eemalda kleepuv)), on võimalik tekstuur renderdada kaamera koordinaadistikus (niinimetatud "[kaamera laotuses \(Camera Mapping\)](#)").

Window (aken)

Renderdatud pildi koordinaadid aknas. See sobib hästi kahe objekti kokkusulutamiseks.

Normal (normaalid)

Kasuta koordinaadina pinnanormaali vektori suunda. See on väga kasulik, loomaks teatavaid eriefekte, mis muutuvad vastavalt vaatenurgale.

Reflection (peegeldus)

Kasutab koordinaatidena peegeldusvektori suunda. See on kasulik peegelduskaardi lisamiseks — ja seda vajad sa keskkonnapildi kasutamisel (*Environment Mapping*).

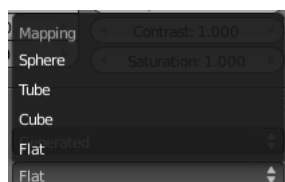
Stress (pinge)

Kasutab erinevust võre käesoleva hetke serva pikkuse ja algse võre servapikkuse vahel. See on kasulik näiteks juhul, kui võre on töötlejate abil deformeeritud.

Tangent (puutuja)

Kasutab tekstuuri koordinaatidena puutujavektorit.

Projektsioon



Projektsioonimenüü Projection

Flat (tasane)

Tasane laotus annab parimaid tulemusi üksikute tasapinnaliste külgede puhul. See annab huvitavaid efekte ka kera puhul, kuid võrreldes kerakujulise laotusega näeb selline objekt lame välja. Külgedel, mis ei asu laotustasapinnal, korratakse tekstuuri viimast pikslit; see tekitab kuubi ja silindri peal triibud.

Cube (kuup)

Kuubilaotus annab enamasti parima tulemuse, kui objekt ei ole liiga looduslik ja laineline (pane tähele "õmblusi" keral).

Tube (toru)

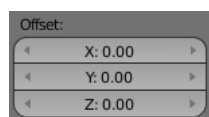
Torulaotus paigutab tekstuuri ümber objekti nagu pudelisildi. Tekstuur on seega silindril välja venitatud. See laotus on loomulikult praktiline pudelitele või muudele ümaratele objektidele siltide ja pealetrükkide lisamiseks. See pole siiski sama mis silindriline laotus ja seega on silindri otste koordinaadid määramata.

Sphere (kera)

Kerakujuline laotus on parim kerakujuliste objektide puhul ja ideaalne planeetide ja muude taoliste objektide tegemiseks. See on tihti ka väga hea looduslike objektide loomiseks. Lisaks tekitab ta huvitavaid efekte silindril.

Koordinaatide pärimine ülemobjektidelt**From Dupli (duplikaatidelt)**

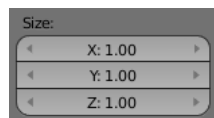
Tippudest, külgedest või osakestest tehtud duplikaadid saavad tekstuurikoordinaadid oma ülematelt.

Vaja teha selgitus**Koordinaatide nihe, skaala ja teisendus**

Nihke paneel Offset

Offset (nihe)

Tekstuurikoordinaate saab nihutada vajaliku väärtuse võrra. Nihke suurendamisel liigub tekstuur üles vasakule.



Suuruse paneel Size

Size (suurus)

Need nupud võimaldavad sul muuta seost tekstuuri enda koordinaatsüsteemi ja valitud koordinaatsüsteemi vahel (genereeritud, UV jne). Täpsemalt: igale tekstuuri teljele saab panna vastama ühe neljast valikust, määrates, millisele laotussüsteemi teljele ta vastab! Sellest järeldub mitu asja:

- 2D-tekstuuride (nagu piltide) puhul on olulised ainult kaks esimest rida, sest sellistel tekstuuridel ei ole Z-komponenti.
- Sa saad 2D-tekstuuri 90 kraadi pöörata, pannes esimese rea (st tekstuuri X-telje) väärtuseks Y ja teise rea (st tekstuuri Y-telje) väärtuseks X.
- Kui sa ei määra mitte ühtegi tekstuuri telge (st kõigi kolme välja väärtus on "none"), saad sa ühtlase ühevärvilise tekstuuri, sest sa kasutad tekstuurist 0 dimensiooni (st ühte punkti või pikslit) ja Blender kordab seda punkti piki kõiki telgi.
- Kui sa määrad ainult ühe tekstuuri telje (st kahe välja väärtus on "none"), saad sa "triibulise" tekstuuri, sest sa kasutad tekstuurist ühte dimensiooni (st ühte pikslite rida) ja Blender venitab seda rida piki teisi telgi.
- Sama juhtub 3D-tekstuuridega (st protseduurilistega), kui ühte telge ei ole määratud. Blender laiendab tasapinnalist "tekstuuri lõiku" piki kolmandat telge.

Ühesõnaga, see kõik on päris keeruline aru saada ja kasutada. Õnneks ei pea sa neid valikuid just tihti muutma, välja arvatud juhul, kui vajad mõnd eriefekti. Igatahes on ainus võimalus nende õppimiseks neid kasutada!

UV-laotus

Kõige paindlikum viis 2D-tekstuuri laotamiseks 3D-objektile on meetod, mida nimetatakse "UV-laotuseks" (*UV mapping*). Selle käigus võtad sa kolmemõõtmelise (X, Y & Z) võre ja lõikad selle lahti kahemõõtmeliseks (X & Y, või nagu me varsti näeme, "U & V") pinnalaotuseks. Pildi värvid projitseeritakse võrele vastavalt laotusele ja need paistavad võre külgede värvadena. Kasuta UV-tekstuurimist siis, kui tahad anda objektidele realistlikumat väljanägemist, kui protseduurilised materjalid suudaksid, ja peenemat detailsust, kui on võimalik saavutada tippe maalides.

UV-kaardi selgitus



Karbi vaatlus



Kaart tasapinnale laotatuna

Parim analoogia UV-laotusest aru saamiseks on kartongist karbi lahti lõikamine. Karp on kolmemõõtmeline (3D) objekt, just nagu kuupvõre, mille sa oma stseenile lisad.

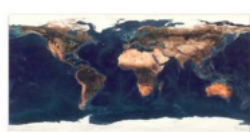
Kui sa võtad käärid ja lõikad kuubi piki (osadid) servi lahti, saad sa karbikartongi lauale tasapinnaliselt laiali laotada. Vaadates laiali laotatud "karbile" ülevalt alla, võime me defineerida suuna U kui suuna vasakult paremale ja suuna V kui suuna alt üles. See pilt on seega kahemõõtmeline (2D). Me kasutame nende "koordinaatide jaoks tekstuuriruumis" tähisteid **U** ja **V**, mitte tavapäraseid **X** ja **Y**. Viimaseid kasutame me (koos **Z**-ga) "3D-ruumi" koordinaatide tähistamiseks.

Kui karp jälle kokku panna, siis teatav UV-punkt paberil liigub kindlasse asukohta (X,Y,Z) karbil. Samamoodi toimib arvuti, kui ta "mähib" 2D-pildi ümber 3D-võre.

UV-lahtilõike käigus pead sa Blenderile täpselt ütlema, kuidas teisendada sinu objekti (antud juhul kuubi) küljed kahemõõtmelisele pildile UV/pildiredaktori aknas. Sul on absoluutne vabadus määrata, kuidas seda täpselt teha. (Jätkates eelmist näidet, kujuta ette, et pärast karbi pinna lauale laotamist lõikad sa ta nüüd väiksemateks tükkideks ja venitad või surud osad tükid mingil viisil kokku ning paigutad nad seejärel mingil viisil laual lebavale fotole.)

Kartograafiline näide

Kartograafid (kaarditegijad) on selle probleemiga tuhandeid aastaid tegelenud. Üks näide kartograafiast on koostada kogu maailma projektsioonkaart. Kartograafias võtame me maakera pinna (kera) ja teeme tasapinnalise kaardi, mille saab kokku lappida kosmosesüstiku kindlaekasse. Me 'lisame' pooluste suunas tühja ruumi või muudame kaardi piirjooni mõnel kindlal viisil:



Merkatori projektsioon



Mollweide projektsioon

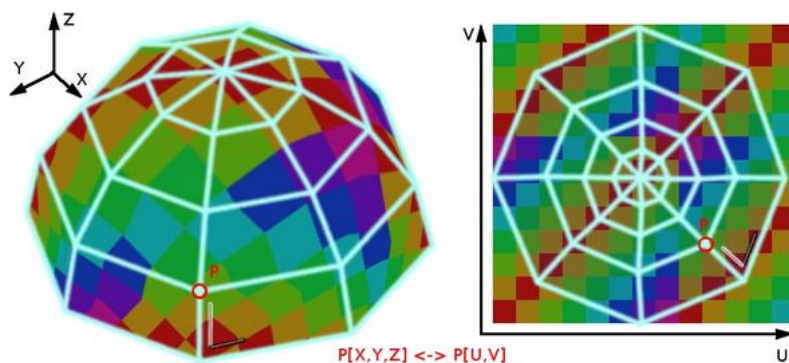


Albers-võrdne projektsioon

Igaüks neist on kera UV-laotuse näide. Kokku on umbes sada laialt kasutatavat projektsioonimeetodit, millest igaühel on omad plussid ja miinused. Blender võimaldab meil teha arvutis sedasama, kuidas aga soovime.

Keerulisematel mudelitel (nagu ülaltoodud maakaart) tekib olukord, kus külgi ei saa lihtsalt lahti 'lõigata' ja pinnale laotada, vaid neid tuleb tasapinnaliseks muutmiseks ka venitada. Nii muutub UV-laotus küll lihtsamaks, aga objektile laotatud tekstuuril võivad tekkida moonutused. (Maad ja riigid, mis on lähemal põhja- või lõunapoolusele, näivad tasapinnalisel kaardil väiksemad kui need, mis on ekvaatori lähedal.)

Poolkera näide



3D-ruum (XYZ) versus UV-ruum (suurendamiseks klõpsa)

Sellel pildil on selgesti näha, et 3D-ruumis valitud külje suurus ja kuju on UV-ruumis teistsugune.

See erinevus on põhjustatud 3D-külje (XYZ) 'venitamisest' (mida tehniliselt kutsutakse laotuseks) 2D-tasandile (st UV-kaardile).

Kui 3D-objektil on UV-kaart, siis on igal objekti punktil lisaks 3D-koordinaatidele X, Y ja Z määratud veel vastavad U ja V koordinaadid. (Punkt P ülaltoodud pildil on näide, kuidas 3D-objekti punkt teisendatakse 2D-pildi punktiks.)

UV-redaktor

UV-koordinaate saab määrata ja muuta UV/pildiredaktoris. Järgnevas ülevaates käsitletakse selle tööriistu. UV-redaktori kasutamist selgitatakse pikemalt järgmistes peatükkides.

Päis

Päisel on mitmed UV-koordinaatidega töötamiseks vajalikud menüüd ja valikud

Vaate menüü *View*

Tööriistad [navigeerimiseks](#), redaktoriga töötamiseks ja asjade esitusviisi määramiseks. Omaduste paneelil on kuvamise seaded ja tööriistad. Kui kasutatakse pilti, näidatakse siin pildi seadeid. Pildiandmete paneeli kasutatakse töös piltidega. Sellel asuvad mitmesugused visualiseerimisvahendid

Valikumenüü *Select*

Tööriistad [UV-elementide valimiseks](#).

Pildimenüü *Image*

Sisaldab valikuid [piltidega töötamiseks](#) ja [tekstuuride maalimiseks](#).

UV-elementide menüü *UVs*

Sisaldab tööriistu [võre lahtilõikeks](#) ja [UV-elementide muutmiseks](#).

Pildivaliku menüü

[Piltidega töötades](#) saab siit valida kasutatava pildi.

[Kinnita pilt](#)

Näitab redaktoris alati praegust pilti, sõltumata valitud objektist ja materjalist.

[Teisenduskeskme valik](#)

Töötas samal viisil nagu teisenduskeskme valik 3D-aknas.

[Sünkroniseeri valik](#)

Hoiab UV ja võre valikud omavahel sünkroonis.

[Valikurežiim](#)

- *Vertex* (tipp)
- *Edge* (serv)
- *Face* (külj)
- *Island* (saar)

[Kleepuva valiku režiim](#)

Kui valiku sünkroniseerimine on välja lülitatud, määrab see menüü, kuidas täpselt UV-tippe valitakse.

[Mõjualaga muutmine](#)

Töötas nagu [Mõjualaga muutmine 3D-aknas](#)

[UV-koordinaatide nakkumine](#)

Samaneb nakkumisele 3D-aknas

[Aktiivse UV-tekstuuri valik](#)

Valib, millist UV-tekstuuri kasutada

Omaduste paneel

Rasvakriit

Töötas samaselt [raskvakriidile](#) 3D-aknas.

UV Vertex (UV-tipp)

Valitud UV-tippude [teisendused](#)

Image (pilt)


Sisaldab valitud [pildi](#) seadistusi

Display (esitus)



Määrab [UV-elementide esitamise viisi](#) ja täiendavad seadistused [tööks piltidega](#).

UV-ruumis navigeerimine

Kogu pilti saab nihutada, vajutades MMB  ja lohistades.

Suurendada ja vähendada saab MMB  üles või alla kerides. Ja nagu 3D-aknas, võib kasutada suurenduse muutmiseks + NumPad või - NumPad.

Töötavad järgmised kiirklahvid (samad valikud on ka vaate menüüs View):

- Suurenda 1:8 8 NumPad
- Suurenda 1:4 4 NumPad
- Suurenda 1:2 2 NumPad
- Suurenda 1:1 1 NumPad
- Suurenda 2:1 ⇄ Shift2 NumPad
- Suurenda 4:1 ⇄ Shift4 NumPad
- Suurenda 8:1 ⇄ Shift8 NumPad
- Näita kõike  Home
- Tsentreeri  NumPad

Esituse valikud

Esituse paneelil Display saab määrata, kuidas UV-elemente näidatakse:

Outline/Dash/Black/White (piirjoon/punktiir/must/valge)

Määrab, kuidas näidatakse UV-servi

Smooth (sile)

Eemaldab servadelt nähtavad sakid

Modified (töödeldud)

Näitab UV-aknas töötlejate rakendamise tulemust

Stretch (venitus)

Näitab, kui suur on külgede kuju erinevus UV-koordinaatides ja 3D-koordinaatides. Sinine märgib väikest moonutust ja punane suurt moonutust. Vali, kas näidatakse nurkade (Angles) või pinna (Area) moonutusi.

UV-laotuse kasutamise eelised

Kuigi protseduurilised tekstuurid (kirjeldatud eelnevates peatükkides) on kasulikud – nad ei korda ennas kunagi ja "sobivad" täpselt 3D-objektidele –, ei ole nad piisavalt paindlikud keerulisemate või reaalseste objektide modelleerimiseks. Näiteks ei näe protseduuriliselt genereeritud inimese nahk kunagi piisavalt loomutruu välja. Kortsud inimese näol või kriimustused auto küljel ei esine täiesti juhuslikes kohtades, vaid sõltuvad mudelist ja tema kasutusest. Käsitsi maalitud pildid või reaalseste objektide fotod annavad meile rohkem võimalusi ja suurendavad tõetruudust. Kui kunstnik kasutab UV-tekstuuri, saab ta detailidel (nagu raamatukaaned, tapeet, vaibad, määratud kohad ja muud detailsed elemendid) määrata igat üksikut pinnapikslit.

UV-kaart määrab, mis osa tekstuurist vastab millisele mudeli hulknurgale. Igale hulknurga tipule lisatakse 2D-koordinaadid, mis määravad, milline pildi punkt sellele kohale kantakse. Neid 2D-koordinaate kutsutakse UV-koordinaatideks (võrdluseks XYZ koordinaadid 3D-ruumis). Nende UV-kaartide loomist nimetatakse "lahtilõikeks", sest see meenutab võre lahtilõikamist ja laialivoltimist 2D-tasandile.

Kõige lihtsamate 3D-mudelite jaoks on Blenderil rida automaatseid lahtilõikealgoritme, mida on lihtne kasutada. Keerulisemate 3D-objektide jaoks pole tavalised kuubi, silindri ja kera lõiked enamasti piisavad. Veelgi täpsema projektsiooni saavutamiseks kasuta lahtilõiget juhtivaid õmblusi. Nii saab lisada tekstuure ükskõik kui keerulistele objektidele, näiteks inimpeale või loomadele. Tihti on need tekstuurid joonistatud pildid, mis on loodud Gimpis, Photoshopis või mõnes teises sulle sobivas pilditöötlusprogrammis.

Mängud

UV-laotus on vajalik ka Blenderi mängumootori ja kõigi teiste mängude jaoks. See on de facto mudelitele tekstuuride lisamise standard. Praktiliselt kõik mängudes leiduvad mudelid kasutavad UV-laotust.

Võre UV-laotus

Esimene samm on võre lahtilõikamine. Seda tasub teha siis, kui sa leiad, et võre on piisavalt valmis ja sa ei plaani enam külgi lisada. Kui sa lisad või tükeldad juba lahtilõigatud mudelil külgi, siis Blender küll lisab need uued küljed laotusele, kuid sul võib olla vaja lahtilõiget täiendada või muuta. Sel viisil on võimalik kasutada UV-tekstuuri võre geomeetria muutmise juhisenä.

Selles peatükis käsitletakse UV-lahtilõiget. Järgmised peatükid räägivad [UV-elementide muutmisest](#), selle järel [UV-laotuste haldamisest](#) ja [piltide lisamisest UV-laotustele](#).

UV-kaartidest

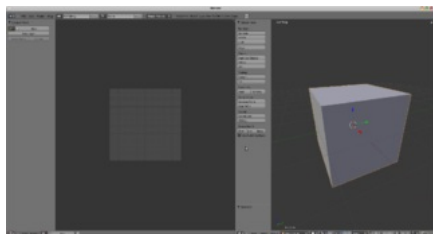
Iga tipp UV-kaardil vastab võre tipule. Jooned, mis ühendavad UV-tippe, vastavad võre servadele. Iga UV-külg vastab võre küljele.

Igal võre küljel võib olla mitu UV-tekstuuri (kaarti). Igale UV-tekstuurile vastab eraldi pilt. Kui sa lõikad võre külje lahti UV-tekstuurile, lisatakse igale võre küljele automaatselt *neli UV-koordinaati*. Need koordinaadid määravad, kuidas pilt või tekstuur küljele paigutatakse. Need on 2D-koordinaadid ning et eristada neid XYZ-koordinaatidest, kutsutakse neid UV-koordinaatideks. Neid koordinaate saab kasutada nii renderdusel kui ka OpenGL-i vahendusel reaajas esitamisel.

Igal võre küljega Blenderis võib olla seotud erinev pilt. UV-koordinaadid määravad, kuidas see pilt küljele laotatakse. Seda pilti saab siis renderdada või reaajas esitada. Et valitud võrele pilte määrata või UV-koordinaate muuta, peab 3D-aken olema "küljevaliku" (*Face Select*) režiimis (Blender 2.4). Nii saab külg olla seotud mitme UV-tekstuuriga. Võre külg tegelase juuksepiiril võib olla seotud nii näo UV-tekstuuri kui ka skalbi/juuste UV-tekstuuriga.

Seda kirjeldatakse pikemalt järgmistes peatükkides.

Alustamine

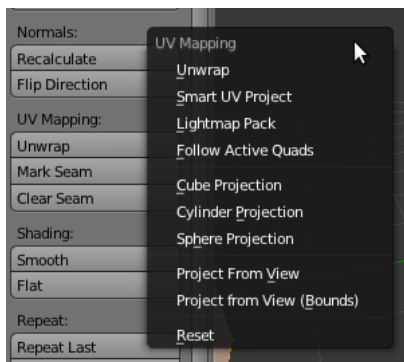


Vaikimisi luuakse võred ilma UV-koordinaatideta. Kõigepealt pead sa võre külgedele laotuse looma ja alles siis saad [seda muuta](#). Võre lahtilõikamine toimub 3D-vaate muutmisrežiimis (*Edit Mode*). Selle käigus luuakse [UV/pildiredaktori aknas](#) üks või mitu UV-saart.

Alustuseks vali kasutaja eelistuste akna nimekirjast töölaua ülaservas UV-redigeerimise [ekraanipaigutus](#) UV Editing. Töölaud jaotatakse kaheks paaniks, millest ühes on UV/pildiredaktor (Shift+f10) ja teises 3D-aken (Shift+f5).

Lülita sisse muutmisrežiim, kus tehakse kõik UV-lahtilõikega seotud toimingud. Võid valida nii tipu-, külje- kui ka serva valimise režiimi.

Töö käik



Lahtilõike meetodi valimine

Lahtilõige ise on lihtne, kuid tal on tohutult erinevaid seadistusi, millest igaüks võib oluliselt muuta tulemust. Kui sa saad seadistuste tähendusest aru, oskad sa võre palju efektiivsemalt lahti lõigata. Tööprotsess on:

1. Kui vaja, märgi õmblused
2. Vali kõik võre osad
3. Vali UV-laotuse meetod menüüst *UV Unwrap* (UV lahtilõige)
4. Seadista lahtilõike parameetrid
5. Lisa testpilt, et näha, kas laotusel on moonutusi. Vaata [Piltide lisamine UV-laotusele](#)
6. Paranda UV-redaktoris laotuse koordinaate. Vaata [UV-laotuse muutmine](#).

Laotuse tüübid

Blender pakub UV-lahtilõikeks mitut viisi. Lihtsamad projektsioonimeetodid kasutavad valemeid, mis projitseerivad 3D-ruumi 2D-tasapinnale, interpoleerides punktide asukohti punkti, telge või pinda läbiva tasandi suunas. Täiuslikumaid meetodeid kasutatakse keerulisemate mudelite puhul ja erijuhtudel.

Lihtsad:

[Kuup \(*Cube*\)](#)

Projitseerib võre kuubi külgedele ja lõikab siis selle lahti.

[Kera \(*Sphere*\)](#)

Projitseerib UV-koordinaadid kerakujulisele pinnale. See annab tulemusi ainult kerakujuliste objektide puhul, nt silmad, planeedid jne.

[Silinder \(*Cylinder*\)](#)

Projitseerib UV-koordinaadid silindrilisele pinnale.

[Vaateprojektsioon \(*Project from View*\)](#)

Loob võrest sellise projektsiooni, nagu see näeb välja 3D-vaates.

Täiuslikumad:

[Lahtilõige \(*Unwrap*\)](#)

Kasulik looduslike objektide jaoks. Laotab võre tasapinnale, lõigates küljed õmbluste kohalt lahti.

[Tark UV-projektsioon \(*Smart UV Project*\)](#)

Jagab võre saarteks külgedevaheliste nurkade järgi

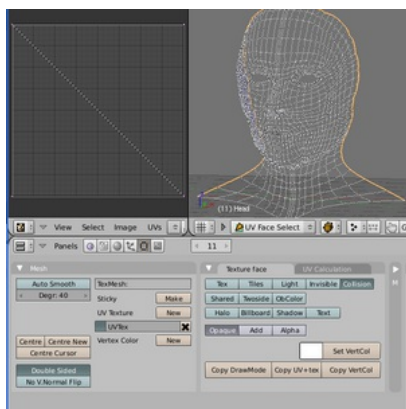
[Valguskaardi pakkimine \(*Lightmap Pack*\)](#)

Eraldab kõik küljed ja pakib need tihedalt UV-ruudustikule

[Järgi aktiivseid nelinurki \(*Follow Active Quads*\)](#)

Jätkab UV-laotust valitud nelinurkadest piki küljeringi

Sa saad alati [UV-koordinaadid jälle algväärtustada](#) – see laotab iga külje tervele UV-võrgustikule, andes seega igale küljele ühesuguse laotuse.



UV-külgede valimine küljevaliku režiimis

Kui sa plaanid kasutada pidevat mustrit moodustavat pilti, siis selline algväärtustatud laotus katab kogu pinna pildi sujuvate kordustega (mida on venitatud vastavalt iga üksiku külje väljanägemisele). Kasuta seda valikut, kui tahad laotuse algväärtustada ja lahtilõikest loobuda (minna tagasi algusesse).

Põhilaotused

Vastavalt objekti põhigeomeetriaale ja sellele, kuidas ta 3D-aknas paistab, üritavad Mesh->UV Unwrap->Cube, Cylinder, and Sphere (võre->UV lahtilõige->kuup, silinder ja kera) UV-algoritmid külg sinu eest nii hästi kui võimalik lahti voltida. Sealjuures on vaade 3D-aknas eriti oluline. Samuti peaksid kuubi suurus ja silindri raadius (muutmisnupud, UV arvutamise paneel *UV Calculation*) olema seatud sellistele väärtustele (Blenderi ühikutes), et terve objekt mahuks nende sisse.

Kuubi, silindri ja kera lahtilõikete jaoks kasutatakse tavaliselt järgmisi valikuid:

Paranda suhet (*Correct Aspect*) UV-koordinaatide arvutamisel arvestatakse pildi külgede suhet. Kui pilt on juba seostatud tekstuuriruumiga, mis ei ole ruudukujuline, võtab projektsioon selle arvesse ja muudab laotust nii, et see paistab õige. Lõika servadel (*Clip to Bounds*) Kõik UV-koordinaadid, mis asuvad väljaspool 0–1 vahemikku, lõigatakse sellesse vahemikku, nihutades nad lähima UV-ruumi küljeni. Skaleeri servani (*Scale to Bounds*) Kui UV-kaart on suurem kui vahemik 0–1, vähendatakse tervet kaarti, et ta mahuks vahemiku sisse.

Kuup

Kuuplaotus (*Cube*) projitseerib võre kuuele eraldi küljele ja loob 6 UV-saart. UV-redaktoris on nad algul kohakuti, kuid neid saab eraldi liigutada. Vaata [UV-laotuste muutmist](#)

Cube Size (kuubi suurus)

Määrab kuubi suuruse, millele võre projitseeritakse

Silinder ja kera

Silindrilisel (*Cylinder*) ja kerakujulisel (*Sphere*) laotusel on samad valikud. Vahe on selles, et silindriline laotus projitseerib UV-koordinaadid otse silindrile, samas kui kerakujuline laotus võtab arvesse kera kõverust, paigutades paralleeljooned üksteisest võrdsele kaugusele.

Tavaliselt laotatakse silinder (toru) tasapinnale, lõigates ta ühelt küljelt pikuti lahti ja painutades laiali. Selle saavutamiseks tahab Blender, et vaade silindrile oleks ülevalt, st toru seisaks 'püsti'. Teistsuguste vaadete korral projitseeritakse toru UV-laotusele teismoodi ja pilt on kaldu. Projektsioonitelge saab siiski ka käsitsi määrata. Sama idee töötab ka kerakujulise laotuse puhul:

On sul veel meeles eespool esitatud kartograafilised projektsioonid? Sa saad sarnase tulemuse, lõigates kera lahti erinevaid vaatenurki kasutades. Kõige tavalisemal juhul lõika kera lahti niimoodi, et poolused oleksid (ekraanil) üleval ja all. Lahtilõike tulemusena loob Blender Mercatori projektsiooniga kaardi. See punkt ekvaatoril, mis on otse sinu suunas projitseeritakse pildi keskele. Vaade pooluselt loob teistsuguse, kuid siiski küllalt tavalise projektsiooni. Kui sa kasutad UV-pildina füüsilist maailmakaarti Mercatori projektsioonis, saad sa kerast hõlpsalt teha planeedi Maa.

Direction (suund)

View on Poles (vaade poolustelt)

Kasuta siis, kui vaatad kera otse pooluse suunast (telg on suunatud otse 3D-vaate "sisse")

View on Equator (vaade ekvaatorilt)

Kasuta siis, kui vaatad kera ekvaatorilt (telg ühtib ekraani vertikaalsuunaga)

View on Object (vaade objektilt)

Võta telje suund objekti koordinaatsüsteemist

Align (joonda)

Määrab, milline telg on vertikaalsuund

Polar ZX (polaarne ZX)

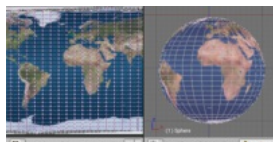
Poolus asub X-teljel:

Polar ZY (polaarne ZY)

Poolus asub Y-teljel

Radius (raadius)

Kasutatava projektsioonisilindri raadius



Mercatori kaardi kasutamine kera lahtilõikel

Projektsioon vaatest

Valik Face->Unwrap UVs->Project from View (kõlg->UV lahtilõige->vaateprojektsioon) laotab küljed UV-kaardile täpselt nii, nagu nad paistavad 3D-vaates. Umbes nagu röntgenpilt objektist või nagu litsuksid objekti UV-kaardile laiaks. Kasuta seda valikut siis, kui sul on modelleeritud objekti UV-tekstuuriks reaalse objekti pilt. Kohtades, kus mudeli küljed on vaate suhtes suure nurga all, venitatakse tekstuur välja.

Valik Project from View (Bounds) (piiratud vaateprojektsioon) teeb sedasama mis eelmine meetod, kuid skaleerib UV-koordinaadid pildi piiridesse.

UV-algväärtustamine

Valik Face->Unwrap->Reset (kõlg->lahtilõige->algväärtusta) 3D-aknas laotab iga külje ühele ja samale pildialale, nagu eelnevalt kirjeldatud. Et laotada kõiki objekti (näiteks kuubi) küljed samale pildile, vali nad ja lõika nad lahti valiku *Reset* (algväärtusta) abil.

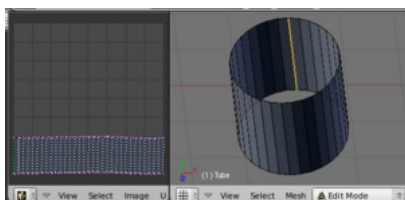
Täiuslikumad lahtilõiked

Lahtilõige õmbluste abil

Paljudel juhtudel annavad kuubi-, silindri- või kera lahtilõikealgoritmid hea tulemuse (eriti pärast sobitamist). Siiski on sul keerukate võrede puhul (eriti kui nad sisaldavad palju vagusid ja eenduvaid detaile) suure tõenäosusega vaja määrata **õmblused**, mis piiravad ja juhivad lahtilõikeprotsessi.

Nagu riietel on õmblused kohad, kus kangatükid on kokku õmmeldud, on laotusel õmblused kohad, kus erinevad pildi alad on kokku liidetud. Lahtilõige teostatakse õmbluste kohalt. Võid mõttes seda meetodit võrrelda apelsini koorimise või looma nülkimisega. Sa teed nahka või koorde seeria lõikeid ja siis koorid selle maha. Seejärel saab naha mõningase venitamise abil sirgeks laotada. Need lõiked täidavad sama funktsiooni nagu õmblused.

Selle meetodi kasutamisel pead sa jälgima, kui palju lahtilõiget venitatakse. Mida rohkem on õmblusi, seda vähem tuleb UV-kaarti venitada, kuid tekstuurimine võib olla tülikam. Üldiselt on kasulik, kui nii õmblusi kui ka venitusi on nii vähe kui võimalik. Ürita peita õmblused sellistesse kohtadesse, kus neid ei ole näha. Kui kasutada 3D-maalimist, on see probleem väiksem, sest projektsioonmaalimisega saab hõlpsalt maalida üle õmbluste – erinevalt 2D-pilditöötlusest, kus eri UV-saarte servade kokkuajamine on väga tülikas.




Lihtne õmblus silindril

Töö käik on järgmine:

1. Määra õmblused. Õmblused tehakse muutmisrežimis, valides kõigepealt vastavad servad ja kasutades siis käsku *Mark Seam* (märgi õmblus).
2. Lõika lahti
3. Paranda õmblusi ja korda
4. Paranda käsitsi UV-laotust. UV-laotuse muutmise kohta vaata järgmist peatükki.

Õmbluste märkimine

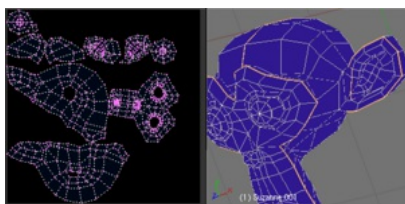
Õmblusi saab märkida muutmisrežimis servavaliku allrežimis. Vali servad, piki mida õmblus paikneb, kasutades Shift RMB , ja vajuta siis **CtrlE** või vali menüüst *Mesh->Edges->Mark Seam* (võre->servad->märgi õmblus). Parempoolsel näitel on õmblus paigutatud silindri kõige tagumisele servale (et teda varjata). Lahtilõige on teostatud vaikeseadistustega. UV/pildiredaktoris on näha, et kõik silindri küljed on kenasti laiali laotatud, otsekui oleks silinder õmbluse kohalt kääridega lahti lõigatud ja siis laiali painutatud.



Oih! Õmblusest ununes üks serv välja

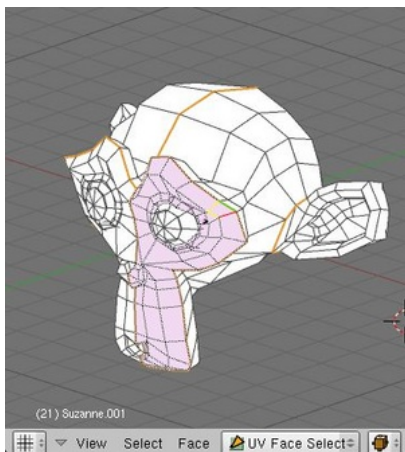
Õmbluste märkimisel võid tulemuse kontrollimiseks kasutada käsku *Select->Linked Faces* (valik->ühendatud küljed) või küljevaliku režimis **CtrlL**. See käsk valib kõik hetkel valitud küljega ühendatud küljed kuni õmbluseni. Kui selle tulemusena valitakse ka küljed, mis asuvad väljaspool soovitud ala, ei ole õmblus järelikult pidev. Õmblused ei pea siiski alati olema pidevad ja piirama lahtilõigatavat ala. Piisab, kui nad aitavad vähendada laotuse külgede väljavenitust.

Et õmblusele serva lisada, lihtsalt vali see serv, vajuta **CtrlE** ja *Mark Seam* (märgi õmblus). Serva eemaldamiseks õmblusest vali see serv, vajuta **CtrlE** ja seejärel *Clear Seam* (eemalda õmblus).



Õmblustega Suzanne

Nii nagu on palju viise kassi nülkimiseks, on ka palju erinevaid võimalusi õmbluste paigutamiseks. Üldiselt katsu mõelda, nagu hoiaksid sa ühes käes objekti ja teises käre ning üritaksid objekti lahti lõigata ja lauale laotada võimalikult vähesel venitamisel ja rebenditega. Pane tähele, et me lisasime õmblused kõrvade välisküljele, et eraldada nende esimene ja tagumine pool. Suzanne'i silmad on omaette alamvõred ja seetõttu laotatakse nad automaatselt eraldi. Õmblus jookseb vertikaalselt piki pea tagakülge nii, et kumbki pea pool laotatakse eraldi.



Külgede valiku režim.

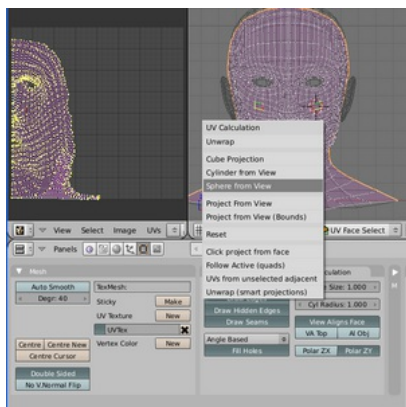
Teine õmbluste kasutamise eesmärk on nende külgede määramine, mida tahetakse laotada. Näiteks pea tekstuurimisel ei ole põhjust tekstuurida pea üla- ja tagaosa, kuna see jääb juuste alla. Nii võid sa teha juustepiirile õmbluse. Kui sa siis lahtilõikamiseks valid mõne näo külje ja seejärel ühendatud küljed, piirduv valik õmblusega ääristatud alaga ja ülejäänud pead ei lõigata lahti.

Kui sa lõikad lahti bilateraalssümmeetrilist objekti nagu pea või keha, paiguta õmblus piki keskjoot. Näiteks lõhesta pea või terve keha

eestvaates otse ülevalt alla. Lahtilõikel võid sa paigutada mõlemad pooled ühele tekstuuriosale nii, et parem ja vasak pool jagavad samu pikseleid. Sel juhul on parem ja vasak pool muidugi identsed.

Lõpuks pea meeles, et sul *ei ole* vaja leida "ühte lahtilõiget, mis töötab kõikjal ja alati." Nagu me edaspidi näitame, saad sa hõlpsalt kasutada sama objekti eri piirkondade jaoks mitut erineva meetodiga lahtilõiget.

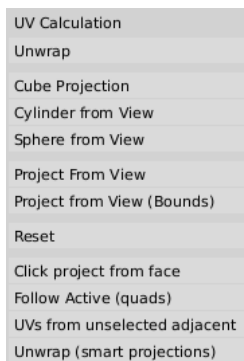
Lahtilõige



Külgede lahtilõige 3D-vaate (**3D View**) menüüst (Blender 2.4)

Kui küljed valitud, on käes aeg võre lahti lõikamiseks. Ava 3D-aknas menüü Face->Unwrap UVs (Külg->UV-lahtilõige) või vajuta kiirklahvi U ja vali *Unwrap* (lahtilõige).

Seda saab teha ka UV/pildiredaktori aknas menüüst UVs->Unwrap (UV-elementid->lõika lahti) või kiirklahviga E. See meetod lõikab kõik küljed uuesti lahti ja kirjutab eelneva töö üle.



Külgede lahtilõike menüü

Käsk Face->Unwrap->Unwrap (külg->lahtilõige->lahtilõige) lõikab valitud objekti küljed lahti 'parima sobituse' meetodil ehk vastavalt sellele, kuidas küljed on omavahel seotud, kuidas nad mahuvad pildile ja kuidas õmblused on valitud külgedel paigutatud. Kui vähegi võimalik, antakse igale küljele oma isiklik tükike pildist, nii et ükski külg ei ole *lõikatud teise külje alla*. Kui kõik objekti küljed on valitud, laotatakse iga külg mingile pildiosale.

Blenderil on kaks lahtilõike arvutamise meetodit. Nende vahel saad valida 3D-vaate tööriistapaneelil.

Angle Based (nurgal põhinev)

See loob võrest hästisobiva 2D-kujutise.

Conformal (sobitatud)

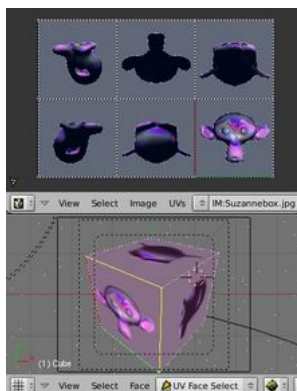
Kasutab LSCM (*Least Squared Conformal Mapping* - vähimruutudel põhinev vastavus) meetodit. Selle meetodiga tehtud UV-lahtilõiked pole tavaliselt nii täpsed, kuid lihtsate objektide puhul töötab ta paremini.

Kui lülitada sisse valik Fill Holes (täida augud), välditakse külgede ülekattet ja võre avad on UV-lahtilõikel õigema kujuga.

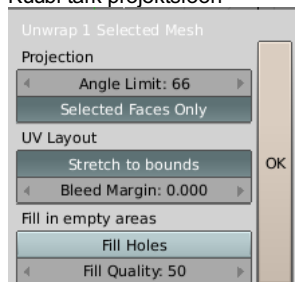
Edasisest aru saamiseks **on ülimalt oluline mõista**, et külje UV-tekstuur võib kasutada ainult *osa*, mitte *tervet* pilti. Samuti seda, et sama pildi sama piirkonda võivad kasutada mitu külge. Külge võib laotada kogu pildi väga väikesele osale.

Tark UV-projektsioon

Valik Face->Unwrap->Smart UV Project (Külg->lahtilõige->tark UV-projektsioon), (vana nimega Archimapper) võimaldab vastavalt sinu poolt määratud seadistustele tekitada võre külgedevaheliste nurkade järgi automaatsed lahtilõikeõmblused. See meetod töötab hästi lihtsate ja keeruliste geomeetriliste kehade puhul, nagu näiteks mehhaanilised objektid ja arhitektuurilised mudelid.



Kuubi tark projektsioon



Targa projektsiooni seadistuste paneel

See meetod analüüsib objekti kuju, valitud külgi ja nendevahelisi suhteid ning loob selle põhjal UV-kaardi, vastavalt sinu määratud parameetritele.

Parempoolisel näitel on "tark lahtilõige" laotanud kõik kuubi küljed kenasti kahes reas UV-kaardile: 3 külge ülemises, 3 külge alumises reas ja kõik küljed ruudukujulised (nagu kuubi küljed peaksid olema).

See tööriist suudab ka keerulisematest mehhaanilistest objektidest teha iseseisvalt väga loogilised ja kenad lahtilõiked.

Automaatset lahtilõiget saab tööriistapaneeli vastavate seadistuse abil täpselt juhtida:

Angle Limit (nurga piir)

Määrab, kuidas küljed grupeeritakse. Suurem väärtus tekitab palju väikesi grappe ja vähendab moonutusi. Väiksem väärtus tekitab küll vähem grappe, aga suurendab võimalikke moonutusi.

Island Margin (Saarte vahe)

Määrab, kui tihedalt UV-saared laotusel paigutatakse. Suurem väärtus lisab saarte vahele rohkem ruumi.

Valguskaart

Valguskaart (*Lightmap*) võtab kõik võre küljed või valitud küljed ja paigutab nad ilma ülekatteta UV-tekstuuri piiridesse. Valguskaarte kasutatakse põhiliselt mängudes, kus pinnale langeva valguse määr lisatakse objektile eraldi tekstuurina. Siin on oluline kasutada võimalikult suurt osa tekstuuri pinnast. See meetod töötab ka siis, kui korraga on valitud mitu võret. Tal on mitmed valikud, mis on kättesaadavad tööriistariiulilt:

Sa võid määrata, kas ühe võrega töötades laotatakse ainult valitud küljed (Selected Faces) või kõik küljed (All Faces).

Valik Selected Mesh Object (valitud võreobjekt) töötab mitme võre korral. Et seda kasutada, vali objektirežiimis mitu võreobjekti, mine muutmisrežiimi ja käivita see tööriist.

Share Tex Space (jaga tekstuuriruumi)

See on kasulik, kui tahad laotada rohkem kui ühte võret korraga. See üritab mahutada kõigi objektide kõik küljed ilma ülekatteta sama UV-tekstuuri piiridesse.

New UV Layer (uus UV-kiht)

Kui laotad mitut võret korraga, siis luuakse iga võre jaoks uus UV-kiht. Vaata [UV-laotuste haldamise](#) peatükki.

New Image (uus pilt)

Loob iga võre jaoks uue pildi (või kui Shared Tex Space (jaga tekstuuriruumi) on sisse lülitatud, siis kõigile võredele ühise pildi).

Image Size (pildi suurus)

Määrab uue loodud pildi suuruse.

Pack Quality (pakkimise kvaliteet)

Kasutab enne keerulisema kastpakkimise algoritmi rakendamist eelpakkimist.

Margin (õhuvahe)

Määrab, kui lähedale UV-saared laotusel üksteisele on pakitud. Suurem väärtus lisab saarte vahele rohkem ruumi.

Järgi aktiivseid nelinurki

Käsk Face->Unwrap->Follow Active Quads (külg->lahtilõige->järgi aktiivseid nelinurki) laotab valitud küljed lahti piki pidevaid küljeringe, isegi kui võre küljed on ebaregulaarse kujuga. Pane tähele, et see ei arvesta pildi suurusega, nii et laotuse pildile mahutamiseks peab sa seda võib-olla vähendama.

Küljepikkuse režiim:

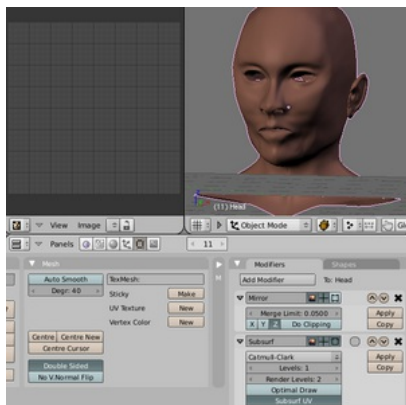
Even (ühtlane)

Paigutab UV-küljed ühtlaste vahemaadega.

Length (pikkus)

Paigutab lahtilõikel UV-küljed vastavalt küljeringi pikkusele võrel.

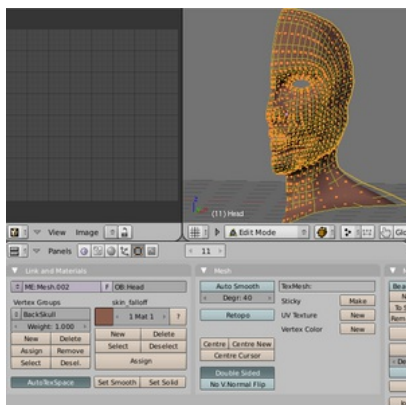
Mitme külje lahtilõige



Alustamine objektirežiimis

Üldreeglina peaksid sa lahti lõikama ainult vajalikud küljed ja tegema seda ühe lahtilõike operatsiooniga. Lahti on vaja lõigata ainult need küljed, millele laotatakse pilt. Ülejäänud küljed võivad kasutada protseduurilisi materjale või tipuvärve. Sa tahad kasutada nii väikseid pilte kui võimalik, seega tahad sa ka lahti lõigata nii vähe külgi kui võimalik. Näiteks kui tegelase keha on alati kaetud riiete või turvisega, ei ole vaja seda lahti lõigata; teine võimalus on see lahti lõigata eraldi kaardile.

Pea tagakülg on alati juustega kaetud ja seda pole vaja lahti lõigata. Kui sa modelleerid tikitud padjaga tooli, siis on vaja lahti lõigata ainult padi, mitte toolijalad. Parempoolsel näitel peame me lõikama lahti ainult ühe näopole, mis vähendab pildi suurust poole võrra. Seega jätame me peegeldamismuutuja sisse. Samuti pole vaja pinnatükeldajat rakendada UV-koordinaatide arvu kahekordistada, vaid jätame ta nii, nagu ta on. Pane tähele, et praegusel hetkel pole võre paneelil veel UV-tekstuuri.



Külgede valimine muutmisrežiimis

Et laotada mitu külge samale tekstuurile, sisene muutmisrežiimis (⇌ Tab) küljevaliku režiimi ja vali soovitud küljed (Blender 2.4).

Parempoolsel näitel on näha, kuidas me oleme peitnud hulga külgi, kõrvad ja pea tagaosa. Me tegime seda tipugruppide abil, valides "BackSkull" tipugrupi ja peites H vastavad küljed vaatest. Seda me tegime sellepärast, et me ei taha neid piirkondi lahti lõigata, ja ei taha ka, et nad jääksid meile ette, kui me hiljem tahame küljevalikurežiimis laotust täiendada.

UV-laotuste haldamine

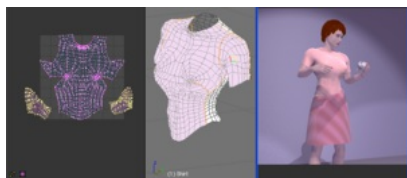
Kui sa oled UV-kaardi valmis saanud, võib sul olla vaja luua samale objektile veel kaarte või kanda see kaart mõnele teisele võrele üle.

UV-laotuse ülekandmine

Sa saad ühe võre UV-laotuse kopeerida teisele sama geomeetria ja tippude järjekorraga võrele. See võib osutuda kasulikuks näiteks siis, kui sa kustutasid kogemata mudeli UV-kaardi, kuid sul on alles varasem salvestatud fail õige UV-laotusega.

Et UV-kaarti üle kanda, vali sihtmärkvõre, seejärel vali klahviga **⇧ Shift** UV-kaardiga võre ning objekti menüüst Object sissekanne Join as UVs (ühenda UV-koordinaatidena). Sihtmärkvõre UV-kaart on nüüd sama nagu lähtevõrel.

Mitme UV-laotuse kasutamine



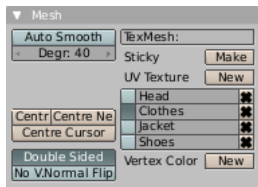
Võre mitme UV-tekstuuriga

Sa saad ühel võrel kasutada rohkem kui ühte UV-laotust. Võre osadele saab teha mitu UV-laotust, luues uusi UV-tekstuure. Esimene UV-tekstuur luuakse siis, kui valid UV-külgede valimise režiimis mõne külje (Blender 2.4). Sa saad UV-tekstuure juurde luua, vajutades muutmiskontekstis *Edit* nuppude aknas võrepaneelil *Mesh* nuppu New (uus) valiku *UV Texture* (UV-tekstuur) kõrval ja lõigates lahti mõne teise võre osa (Blender 2.4). Need võre küljed seotakse uue UV-tekstuuriga. Samas eelnevalt lahti lõigatud küljed jäävad seotuks eelmise UV-tekstuuriga. Pane tähele, et kui sa lõikad sama külje lahti kaks või enam korda (iga kord erinevale UV-tekstuurile), määrab selle külje värvi kõigi nende tekstuurikihtide läbipaistvuste kombinatsioon.

Parempoolsel näitel on modelleeritud pluus. Võrel on õmblused paigutatud reaalse pluusi eeskujul, nagu näha keskmisel pildil (UV-külgede valimise režiimis). Et järgida lõike mustrit, lõigati pluusi esitükk lahti ning paigutati see pöörates ja selle mõõtkava muutes UV/pildiredaktori akna keskele. Seejärel nihutati see ühele küljele, lõigati lahti vasak ja parem käis ning pöörati ja muudeti nende laotuste mõõtkava. Nüüd vali 3D-aknas üks külg igast riidetükist ja seejärel *Select->Linked Faces* (vali->ühendatud küljed) ning UV/pildiredaktor näitab kõiki neid osasid (nagu näha vasakpoolsel pildil). Seejärel saad sa töötada kõigi selle UV-laotuse osadega korraga. Näites on näha, kuidas kõik osad on nihutatud maalimiseks pildiväljale. Igal juhul näed sa, et lõige mahub ilusti ruudukujulisele riidetükile.

Käiste tagakülgede ja seljatüki jaoks tehti uus UV-lõige, vajutades nuppu New (uus) võrepaneelil *Mesh* ja korrates lahti lõikamist. Riide värvi määravad kaks pilti: üks esi- ja teine tagaosa. Antud juhul on osad küljed laotatud esimesele, osad teisele tekstuurile.

UV-tekstuuride nimekiri



Parempoolsel pildil oleval võrepaneelil *Mesh* on näha nimekirja kõigist selle võre jaoks loodud UV-kaartidest. Sealsamas saab luua uusi UV-tekstuure (tulevaste lahti lõigete kohatähteks).

Uue UV-tekstuuri lisamiseks vajuta nuppu + ja olemasoleva kustutamiseks nuppu -. Võrega seotud UV-laotuse kustutamine kustutab kogu vastava võre osa lahti lõikamisega seotud informatsiooni. Seega kasuta neid tööriistu ettevaatlikult. Sind on hoiatatud.

Igal kaardil on valikunupp. Vajuta kaamera ikoonile, kui tahad, et seda tekstuuri renderdataks. Tekstuuri nime saab muuta, valides tekstuuri ja muutes väljal Name (nimi) olevat teksti. Valitud UV-kaarti näidatakse UV/pildiredaktori aknas. Antud näites on tegelase jaoks loodud mitu UV-kaarti, kusjuures riietuse kaart on valitud.

Pane tähele, et igat tekstuuri saab siduda eri UV-tekstuuriga. Vaata tekstuuripaneeli kohta täpsemalt [tekstuuride laotus](#) peatükist.

UV-laotuse muutmine

Pärast lahtilõiget pead sa tõenäoliselt UV-laotust korrastama, et saavutada selline loogiline paigutus, mida saab kergesti tekstuurida ja maalida. Muutmise eesmärgid on:

- Ömmelda osad tükid (UV-kaardid) tagasi kokku
- Pildil kasutamata ruumi hulga minimeerimine
- Suurendada 'külgi', millele sa tahad rohkem detaile lisada
- Suurendada ja parandada väljavenitatud 'külgede' kuju
- Vähendada 'külgi', millel on liiga palju detaile ja mis näivad liiga kirjud

Mida vähem on kasutamata ruumi, seda rohkem pildi piksleid osalevad UV-tekstuuris ja võre pind on parema detailsuse ning väljanägemisega. UV-külje suurus võib olla ühest ainsast pikslist (pildipunktist) kuni terve pildi suuruseni. Tõenäoliselt tahad sa esmalt teha suuremad ümberkorraldused ja seejärel sättida laotuse pisemaid detaile.

UV-elementide valimine

Valimise tööriistad on kättesaadavad valikumenüüs Select, päiseribal ja allpool toodud kiirklahvide vahendusel:

Border Select (piirdkastiga valik) ; B

Kasutab ristkülikukujulist raami UV-elementide valimiseks.

Select/Deselect All (vali kõik/tühjenda valik) ; A

Valib kõik UV-elementid või tühjendab valiku. Kohe pärast lahtilõikamist tahad sa ilmselt valida kõik UV-tipud, et neid pöörata, suurendada/vähendada või nihutada.

Linked UVs (ühendatud elemendid);, CtrlL

See valib kõik UV-elementid, mis kuuluvad samasse UV-kaarti. Meenuta, et iga alamvõre ja õmblustega eraldatud võre osa moodustab ühe UV-kaardi (analoogselt riidetükkidega). Ühendatud elementide valimine toimib samamoodi kui analoogne meetod 3D-aknas. See valib kõik UV-elementid, mis on 'ühendatud' juba eelnevalt valitud elementidega.

Pinned UVs (kinnitatud UV-elementid) ; ⇧ ShiftP

UV-elemente on võimalik paika kinnitada nii, et neid ei muudetak järgnevate UV-lahtilõikamiste käigus. See menüüvalik valib kõik kinnitatud elemendid. Vaata lõiku "Kinnitamine"

Border Select Pinned (kinnitatud piirdkastiga valik) ; ⇧ ShiftB

Kasuta riskülikukujulist raami kinnitatud UV-elementide valimiseks.

Unlink Selection (ühenda valik lahti) ; AltL

Lõikab valitud UV-elementid olemasoleva kaardi küljest lahti. Pärast seda käsku jäävad valituks ainult need UV-elementid, mis kuuluvad terveniisti selekteeritud külgedele. Nagu nimest järeldub, on see kasulik selleks, et lõigata külgi lahti ja nihutada neid mujale. Kiirklahv on sama, mis võre eraldamise (*Separate*) käsul.

Valikurežiimid

Valiku sünkroniseerimise päisenupu Sync Selection sisselülitamisel järgib võre elementide valik 3D-aknas UV-elementide valikut UV-redaktoris. Vaikimisi on see välja lülitatud. Need kaks režiimi muudavad oluliselt seda, kuidas toimub elementide valik ja ümberpaigutamine UV-redaktoris.

Kui kleepuv valik (Sync Selection) on **väljas**: UV-redaktoris näidatakse ainult neid külgi, mis on 3D-aknas valitud. Võimalikud on järgmised valikurežiimid:

- Vertex (tipp)

Valib üksikuid tippe

- Edge (serv)

Valib servi

- Face (külg)

Valib külgi

- Island (saar)

Valib ühendatud külgede grupe

Antud režiimis on nähtav kleepuva valiku menüü Sticky Selection Mode. Sellega määratakse, kuidas UV-elemente valitakse:

Shared Vertex (ühine tipp)

Valitakse UV-elementid, millel on ühine võretipp – isegi siis, kui nad asuvad laotusel eri kohtades.

Shared Location (ühine asukoht)

Valitakse UV-elementid, mis on laotusel samas kohas ja millel on ühine võretipp. See on vaikerežiim ja töötab enamasti kõige paremini.

Disabled (välja lülitatud)

Lülitab kleepuva valiku välja. Kui sa liigutad selles režiimis UV-elementi, saad sa neid üksteisest eraldada, sest igal küljel on omad elemendid.

Kui kleepuv valik (Sync Selection) on **sees**, siis on võimalikud järgmised valikurežiimid:

- Vertex (tipp)
- Edge (serv)
- Face (külg)

Selles režiimis käitub valik teistmoodi. Kui valida UV-tippe või -servi, siis käitub ta nagu ühise tipu režiim Shared Vertex ülalpool. Kui valida UV-külgi, siis ta käitub nagu väljalülitatud kleepuv valik Sticky Selection ülalpool.


UV-elementide ümberpaigutamine

UV-elemente saab:


- Nihutada G
- Pöörata R
- Skaleerida S

Samuti saab neid peita ja nähtavaks muuta, kasutades vastavalt kiirklahve H ja AltH – täpselt nagu muutmisrežiimis.

Telje lukustamine

Teisendusi saab lukustada ainult ühele teljele, vajutades pärast teisendustööriista valimist X või Y. Samuti, kui hoida all MMB , lukustatakse teisendus kas X- või Y-teljele.

Teisenduste keskpunktid

UV-redaktoris on 2D-kursor. Selle asukohta saab muuta, vajutades LMB . Sa saad kursori täpse positsiooni sisestada ka seadistuste paneelil *Properties*. Vaikimisi väärtuste vahemik on 0-st 256-ni, kusjuures nullpunkt on vasakus alumises nurgas. Lülitades sisse valiku Normalized (normaliseeritud) koordinaatide alljaotuses Coordinates, on võimalike väärtuste vahemik 0-st 1-ni.

2D-kursoril saab lasta soovi korral nakkuda lähima piksli või valitud elemendiga, valides menüüst UVs nakkumise alammenüü Snap.

Teisenduspunkti võib määrata kas:

- piirdkasti keskpunkti
- mediaanpunkti
- 2D-kursori asupaika

Mõjualaga muutmine

UV-redigeerimisel on võimalik kasutada mõjuala. Selle juhtimine toimub samamoodi nagu 3D-aknas. Vaata täieliku kirjeldust peatükist [mõjualaga muutmine 3D-keskkonnas](#).

Nakkumine

Nakkumine UV-redaktoris on samuti sarnane [nakkumisele 3D-keskkonnas](#), kuid seda ainult UV-tippudega nakkumisel. Seevastu valik Snap to Pixels (nakkumine pikslitega) menüüst UVs liigutab UV-koordinaadid lähimate pikslite keskpunktidesse niipea, kui UV-redaktoris pilt laaditakse.

Menüü UVs nakkumise alammenüüst Snap leiab veel tööriistu:

Snap Pixels (nakkumine pikslitega)
Liigutab valitud tipud lähima piksli asukohta
Snap to Cursor (nakkumine kursoriga)
Liigutab valitud tipud 2D-kursori asukohta
Snap to Adjacent Unselected (nakkumine selekteerimata naabriga)
Liigutab valitud tipud valimata naabri asukohta

Keevitus ja joondamine

Keevitustööriist Weld (W1) liigutab valitud UV-tipud nende asukohtade keskmisesse punkti

Joondajad Align (W2, W3 ja W4) paigutavad UV-tipud ühele joone kas X, Y või automaatselt valitud teljel.

Peegeldus

Elemente saab peegeldada Y- või X-teljelt. Menüü UVs peegelduse alammenüüst Mirror saab valida Mirror X (peegelda X) või Mirror Y (peegelda Y) .

Võid kasutada ka kiirklahvi CtrlM ja selle järel klahve X või Y või hoida all MMB  ja lohistada hiirt peegelduse suunas.

Nõelumine

Nõelumine Stitch (V) ühendab sellised valitud UV-tipud, millel on ühised võretipud. Sa saad piirata maksimaalse õmblusdistsantsi tööriista seadistustest (*Tool Settings*), aktiveerides valiku Use Limit (kasuta piirangut) ja määrates Limit Distance (piirav pikkus) väärtuse.

Venituse minimeerimine

Venituse minimeerimine Minimize Stretch (CtrlV) vähendab UV-elementide väljavenitust nurkade minimeerimise abil. Põhijoontes muudab see UV-laotuse lõdvemaks.

Külgede peegeldamine ja UV-pööre

Mäletad, et UV-tekstuuri orientatsioon on iga külje jaoks suhteline ja omaette? Sa võid näiteks avastada, et pilt on tagurpidi või ühe külje peale kallutatud. Sel juhul kasuta 3D-aknas külgede valimise režiimis valikut Face->Rotate UVs (külg->pööra UV koordinaate), mis pöörab külje UV-koordinaate 90 kraadi kaupa.

Valik Face->Mirror UVs (kõlg->peegelda UV-koordinaadid) pöörab pildi ümber nagu pannkoogi pannil, näidates 'tagurpidi' pilti.

Kinnitamine

Mudelit lahti lõigates on mõnikord kasulik "lukustada" osad UV-tipud, et see osa laotusest säilitaks oma kuju ja/või asukoha.

Tipu kinnitamiseks tuleb valida UV-tipp ja seejärel menüüst UVs kinnitamise sissekannepin või vajutada kiirklahvi P. UV-tippu saab vabastada (Unpin) kiirklahviga AltP.

Kinnitamine on kõige kasulikum looduslike objektide UV-laotuse lahtilõikamisel (*Unwrap*). Näiteks kui sa modelleerid [peegeldaja](#) (Mirror Modifier) abil sümmeetrilist objekti. Mõned UV-tipud peegeldusteljel on ühised esialgsetele ja peegeldatud elementidele. Sa võid need UV-tipud, mis asuvad peegeldusjoonel, kinnitada ja joondada siis X-teljele, sel juhul jäävad nad järgnevatel lahtilõigetel paigale.

Kinnitamine töötab suurepärast ka interaktiivse lahtilõikajaga. Kui sa kinnitad kaks või rohkem UV-tippu ja interaktiivne lahtilõikaja on sisse lülitatud, siis nende tippude liigutamisel lõigatakse mudel jooksvalt uuesti lahti. See aitab mahutada UV-saart kindlasse piirkonda.

UV-laotuse optimeerimine

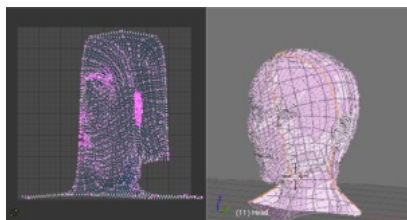
Pärast mudeli lahtilõikamist (võimalik, et õmbusi kasutades) võib esialgne UV-laotus olla küllaltki kaootiline ja segane. Sul on vaja teha läbi järgmised tegevused: UV-laotuse õigetpidi paigutamine, UV-kaartide ümberpaigutamine, erinevate kaartide kokkuvõttelamine.

Järgneb töö lahtilõikamise käigus loodud UV-kaartidega. Kui sa lisad külgi või tükeldad juba olemasolevaid lahtilõigatud võrel, lisab Blender need uued küljed ise UV-laotusele. Sel viisil on võimalik kasutada UV-tekstuuri võre geomeetria muutmise juhena.

Kui sa UV-elemente ümber paigutad, pea meeles, et kuigi töölaud on terve aken, kantakse UV-laotuse joonisele ainult need elemendid, mis on ruudustiku sees. Nii saad sa elemente töötamise ajal kõrvale tõsta. Lisaks on iga UV-lahtilõige iseseisev koordinaatide hulk.

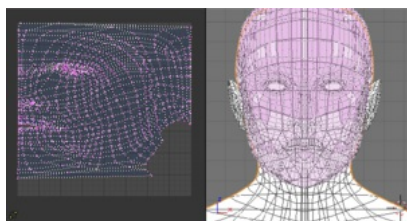
Sa saad nad üksteise peale paigutada ja nad paistavad üksteisest läbi (alumine paistab läbi ülemise). Et valida ühte karti, vajuta ühel UV-tipul RMB ja vali seejärel kõik temaga ühendatud elemendid menüüst Select->Linked UVs (valik->ühendatud UV-elementid) (CtrlL). Piirdkasti ei saa kasutada, sest see valib tipud mõlemalt kaardilt.

UV-kaartide kombineerimine



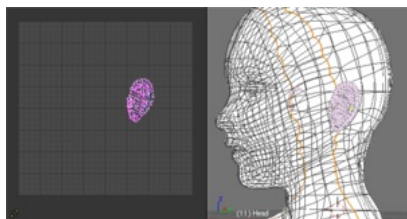
Halb laotus: vaata kaela ja kõrva

Väga tihti, kui sa oled objekti lahti lõiganud (nagu antud näites inimese näo), on laotus 'põhijoontes õige', aga mõned võre osad ei ole õigesti lahti lõigatud või on väga segased. Paremal toodud pildil on näha inimese näo esmane lahtilõige kerakujulise lõikealgoritmiga (*Unwrap from sphere*). Probleemsed kohad on kõrv, mis on täielik UV-tippude ja joonte rägastik, ja kael, mis on välja venitatud ja alt kokku volditud. Selle käsitsi puhastamine on liiga suur töö.



Ainult näo lahtilõige, ilma kõrva ja kaelata

Me teame, et kõrva saab kenasti lahti lõigata külgsuunise projektsiooni ja kaela silindrilise lahtilõike abil. Seega lahendame oma probleemi, lõigates objekti eri osad (näo, kõrvad jne) lahti erinevate meetoditega. Iga osa jaoks valime sellise arvutusmeetodi, mis annab just nimelt tema puhul parima tulemuse. Alustagem siis – valime kõigepealt ainult "näole" kuuluvad küljed, lõikame nad kerakujulise algoritmi (*Sphere*) abil lahti ning skaleerime ja pöörame nii, et nad enam-vähem loogiliselt kataksid pildi pinna UV/pildiredaktori aknas.

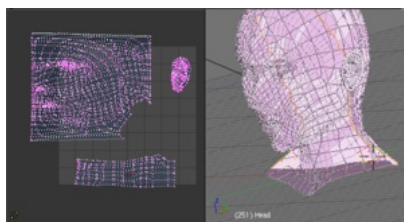


Kõrva lahtilõikamine külgsuunise projektsiooniga

Olles näo paigutusega rahul, pöörame nüüd tähelepanu kõrva lahtilõikamisele. Kõigepealt tühjendame valiku (näole vastavad küljed). Nende UV-koordinaatide pole UV/pildiredaktoris enam näha, kuid nad on seal alles, lihtsalt peidetud. (Et selles veenduda, võid valida 3D-aknas mõned küljed ja nad ilmuvad ka UV/pildiredaktoris.)

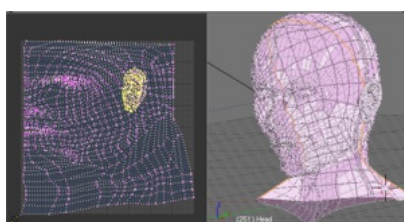
Kõrva lahtipakimiseks valime 3D-aknas ainult "kõrva" moodustavad küljed. Vastavate külgede valimiseks on võimalik kasutada tipugruppe. Samuti on lihtne valida alamvõresid, sest nad ei ole ühendatud ülejäänud võrega. Ühendatud tipude (*Linked Vertices*) valiku kasutamisel valime me terve alamvõre. Ühesõnaga, kuna sa töötad muutmisrežiimis, on võimalik kasutada kõiki sealseid valimise võimalusi.

Nüüd lõikame kõrva uuesti lahti, kasutades külgvaates projitseerimise algorütm (*Project*), teisendame mõõtkava ning pöörame seda pisut (seda selgitame järgmises lõigus) ja paneme selle pildi serva. Sa või seda protsessi teha korduvalt, kasutades erinevaid UV-lahtilõike meetodeid. Iga lahtilõike meetod paigutab lihtsalt UV-tipud erinevatesse kohtadesse. Kasuta iga osa jaoks seda meetodit, mis annab kõige paremini ülejäänuga sobiva ja tekstuuri maalimiseks kõige loogilisema paigutusega laotuse.



Kokkupandud UV-kaardid

Kui kõik võre osad on sobivate meetoditega lahti lõigatud, peaks su tulemus olema umbes samane paremal toodud näitele. Kõigil võre osadel on UV-kaardid ja kõik need kaardid on paigutatud ühisele UV-tekstuuri kaardile. Palju õnne! Siit edasi on vaja lihtsalt tükid kokku "nõeluda" (seda kirjeldatakse järgmises peatükis), et saada ühest tükist koosnev UV-kaart.



UV-kaardid, mis on kokku sobitatud ja nõelutud

Kui sa oled sobitamise ja nõelumise lõpetanud, on sul ühtne UV-kaart, nagu näidatud parempoolsel pildil: see on paigutatud nii, et ühe pildiga saab katta (või ühele pildile maalida) kogu detailset tektuuri vajava võre. Täpsemad juhised, kuidas seda teha, on toodud järgmises peatükis. Antud lõigu mõte on näidata sulle lõppeesmärki. Pane tähele, et näites toodud võre on peegeldatud piki Z-telge ja seega on näo parem pool virtuaalne ehk vasaku poole täpne koopia. Seega on olemas ainult üks kogum UV-koordinaate. (Kui on vaja realistlikumat tulemust, tuleks peegeldamise töötleja (*Mirror Modifier*) rakendada, moodustades tervikliku sümmeetrilise pea. Seejärel on võimalik teha näo pooled füüsiliselt erinevaks, muutes ainult ühte poolt ja mitte teist. Lahtilõikamisel tekib siis terviklik kogum UV-koordinaate (mõlema poole jaoks) ja näo pooli võib veidi erinevalt maalida – mis on realistlikum.)

Keskmine saare mõõtkava

Kasutades tööriista Average Island Scale (keskmine saare mõõtkava), kiirklahv CtrlA, suurendatakse/vähendatakse kõiki UV-saari nii, et nende mõõtkava oleks ligilähedaselt sama.

Saarte pakkimine

Tööriist Pack Islands (paki saared), kiirklahv CtrlP, skaleerib esmalt kõiki saari võrdsel määral ja seejärel liigutab neid, üritades täita UV-ruumi võimalikult tihedalt. See tööriist on oluline tekstuuripinna ökonoomseks kasutamiseks.

Piiramine pildiaärtega

Kui valik Constrain to Image Bounds (piira pildi äärtega) on sisse lülitatud, ei ole UV-tippe võimalik UV-vahemikust 0–1 väljapoole nihutada.

Grab/Move	G
Rotate	R
Scale	S
Weld/Align	W
Mirror...	M

UV-teisenduste menüü

Iteratsioonid ja täiustamine

Meil kui tavalistel inimestel ei tule asjad enamasti "esimesel korral õigesti välja." Algset ideed tuleb üha täiendada ja oma loomeprotsessi uuesti korrata, kuni jõuame maagilise eesmärgini nimega "Valmis". Tarkvaraarenduses nimetatakse seda spiraalimeetodiks.

Rakenduslikus arvutigraafikas liigume me modelleerimiselt tekstuurimisele, tekstuurimiselt animeerimisele ja seejärel tagasi algusesse, et teha võres mõned muudatused, teha uus UV-laotus, täiustada animatsioone või lisada mõni luu. Leides siis, et võrele on vaja külgi juurde ja minnes tagasi modelleerimise juurde jne. Niimoodi me muudkui tiirutame, kuni raha, aeg või kannatus otsa saavad või kuni me mõnel haruldasel juhul tulemusega rahule jääme.

Laotuse täiustamine

Täiustamine osutub vajalikuks siis, kui me vaatame valmis tegelast ja avastame, et mõni koht nõuab suuremat detailsust. Näiteks võib olla vaja lisada silmaümbrusesse kanavarbad või vestile firmamärk. Kui sa siis hakkad pilti muutma, avastad sa, et vajalike detailide maalimiseks ei ole piisavalt piksleid.

Sinu ainus võimalus on siis vastavat UV-külge suurendada. Kasutades venituse minimeerimist või skaleerimist, teed sa küljed silmade ümber või rinnal suuremaks, andes vastavatele aladele rohkem piksleid, võttes samas piksleid (detailsust) vähemaks teistelt kohtadelt (näiteks pea tagaküljelt). Pärast UV-laotuse täiustamist pead sa muutma pilti, et ta näeks õige välja ja sisaldaks vajalikke detaile.

Tekstuuride taaskasutamine

Teine oluline asi, millele mõelda, on ressursside kokkuhoid. Iga pilt laaditakse arvuti mällu. Kui sa saad sama pilti kasutada mitmete võrede jaoks, hoiab see mälu kokku. Nii näiteks võib sul olla üks ja sama ühine 'näo' pilt, mida sa kasutad erinevatel tegelastel, nende erinevaks tegemiseks aga muudad võre UV-laotust ja näo kuju või lisad aksessuaare (päikesepriillid).

Sul võib olla üks "kulunud sinise teksariide" tekstuur ja sa kasutad sama pilti eri tegelaste jalgade lahtilõikamisel. Kasulik on omada universaalset nahatekstuuri ja kasutada seda tegelaste käelabade, käte, jalalabade, jalgade ja kaela jaoks. Kui sa modelleerid fantaasiamööka, piisab väikesest mõõgatera osa pildist. Seejärel sa algväärtustad mõõga külgede lahtilõike (*Reset Unwrap*), nii et piki kogu mõõgatera kasutatakse sedasama pilti.

Piltide kasutamine

Varem või hiljem tahad sa kasutada oma mudelil pilditekstuuri. Kui sa kasutad välist pilditöötlusprogrammi, on sul vaja täpselt teada, millisesse võre kohta sa joonistad. Samuti võib sul olla vaja katsetada oma UV-laotust testpildiga. Selles peatükis näidatakse, kuidas saab eksportida oma UV-laotuse joonist ja kuidas saab UV-redaktorisse pilte laadida.

UV-laotusjoonise eksportimine

Et anda UV-tekstuuri joonistavale kunstnikule ülevaade võre laotusest, on Blenderis UV-külgede laotuse salvestamise tööriist *Save UV Face Layout* (asub UV/pildiredaktori menüüs *UVs->Save UV Face Layout* (salvesta UV-külgede laotus)). Võimalik on salvestada valitud objektist pilt kas Targa (.tga), EPS-i või SVG formaadis.

See pilt kujutab endast objekti UV-laotuse täpset joonist (pinnalaotust). Selle tööriista käivitamisel avaneb failisirvi aken järgmiste salvestusvalikutega:

All UVs (kõik UV-d)

Kui see on välja lülitatud, kantakse joonisele ainult valitud UV-küljed.

Format (formaad)

Määrab pildifaili formaadi.

Size (suurus)

Määrab pildi suuruse pikslites. Pilt on alati ruudukujuline.

Wire (traatjoon)

Määrab külgi piiritlevate joonte paksuse.

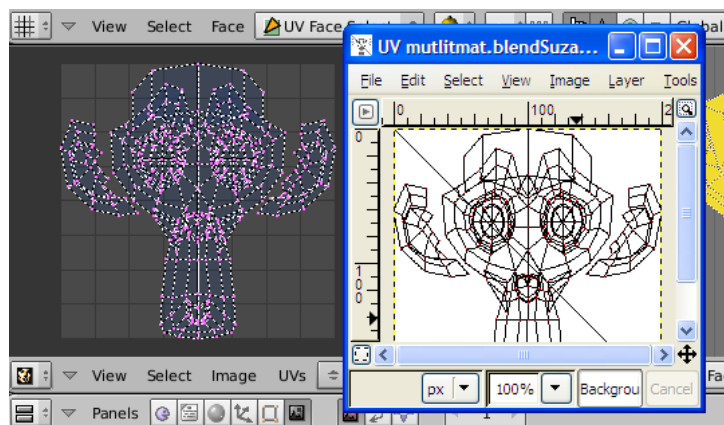
Fill Opacity (värvi läbipaistvus)

Määrab joonte värvi läbipaistvuse.

Pildi moodustavad need UV-külgi piiritlevad jooned (UV-servad), mis asuvad UV-laotuse pildivälja sees. Väljaspool piire asetsevad servi, isegi kui need on valitud, pildile ei kanta.

Kunstnik saab kasutada seda pilti joonistusprogrammis läbipaistva kihina, mis näitab, kuidas tekstuuri eri osad objektile laotuvad. Alltoodud näitel on tagaplaanil Blender ja esiplaanil Gimp; viimases on pooleli töö tekstuuri ja juhisenähtuse kasutatud salvestatud laotust. Pane tähele, et targa-failiformaat toetab alfakanalit (läbipaistvust), seega saad sa maalida ka võre läbipaistvaid alasid.

Seda, kuidas kasutada pilte tekstuuridena, vaata [pilditekstuuride](#) peatükist.



Laotuse kasutamine GIMPis

UV-tekstuuride kasutamine

UV/pildiredaktor võimaldab laotada tekstuure otse võre külgedele. 3D-vaates on näha objekt, millele tekstuuri laotatakse. Kui sa määrad lisaks selle akna esitusviisiks tekstuuritud vaate (*textured shading*), siis näed kõiki UV/pildiredaktori aknas tehtud muudatusi koheselt 3D-vaates ja vastupidi.

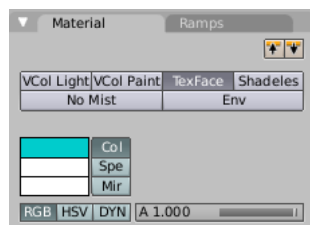
Sa saad pilte muuta ja laadida ja isegi Blenderi mängumootoris mängida, kasutades tegelasi ja objekte UV-tekstuuridega, ilma materjalideta. 3D-vaade näitab ka selliseid objekte. Seda sellepärast, et 3D-aknas ei toimu mitte 'tegelik' renderdus, vaid ainult objekti jooksev esitamine ekraanil. Kui sa aga lisad objektile UV-tekstuuri ja seejärel renderdad, siis tekstuuri vaikimisi näha ei ole.

Selleks, et seda pilti ikkagi renderdada, on vaja

1. luua selle objekti jaoks materjal ja
2. määrata, et Blender kasutaks renderdamisel objekti külgedel UV-tekstuure.

Materjali loomiseks tuleb varjutuskontekstis vajutada uue materjali lisamise nuppu Add New (Blender 2.4).

Et Blender kasutaks renderdamisel UV-tekstuuri, on kaks meetodit: õige meetod ja kiire meetod.



Materjali paneel Material

sisselülitatud küljetekstuuride nupuga Face Textures.

- Kiire meetod

Kiire meetod on lülitada sisse küljetekstuuride materjal (*Face Textures Material*), nagu pildil näidatud. Selleks vajuta nuppude aknas F5, et näidata varjutusnuppe (*Shader Buttons*). Sama akna materjali valikutes vajuta uue materjali lisamise nuppu ADD NEW.

Valikute paneelil lülita sisse Face Textures (küljetekstuurid). See on kiire meetod, mis aga lihtsuse huvides ignoreerib normaalset renderdussüsteemi. Muuhulgas ei saa antud juhul renderdamisel kasutada ei läbipaistvust ega õiget varjutamist.

- Õige meetod

Tekstuurikanalite paneelil F6 (näidatud ülalpool) vajuta *Add a New Texture* (lisa uus tekstuur), määra tekstuuri tüübiks pilt ja laadi soovitud pildifail. Alljaotuses *Mapping* (vastavus), vali menüüst *Coordinates* (koordinaadid) *UV* (UV-koordinaadid) ja seejärel sobiv UV-kiht.

Alljaotuses mõju (*Influence*) veendu, et see pilt oleks seostatud värviga (*Color*) (tekstuuri vaikimisi seos ongi värv; UV-tekstuuri vaikimisi nimi on "UVTex"). Kui pildil on läbipaistvuskanal (*alpha*) ja sa tahad seda kasutada, vajuta pildi seostamise paneelil *Map Image* nuppu *UseAlpha* (kasuta läbipaistvust).

Detailse pilditekstuuride kasutamise õpetuse leiad [pilditekstuuride](#) peatükist.

Renderdamiseks peab objektile olema materjal

Sa saad Blenderis määrata UV-tekstuure ja näha tulemusi 3D-aknas (tekstuuritud esitusrežiimis) ilma objektile materjali lisamata. Renderdamisel on aga ilma määratud materjalita objektide pind ühtlaselt hall. Kui pilti pole laaditud, on pind ühtlaselt must. Kui sa ei loo pilditekstuuri või ei lülita sisse küljetekstuure (Face Texture), renderdatakse objekt vastavalt protseduurilise materjali sätetele.

Piltide laadimine ja salvestamine

UV-redaktoris on võimalik võre eri külgedele määrata erinevad tekstuurid. Selleks on kõigepealt vaja vastavaid pilte. Pildimenüüst Image saad sa failisirvija abil pilte avada. Kui soovid kasutada selliseid pilte, mis juba on Blenderi failis olemas, vajuta päises sirvimise nuppu Browse või loo uus tekstuur, vajutades nuppu New (uus).

Kui .blend-faili poolt kasutatav väline pildifail muutub või Blenderi töötamise ajal (näiteks meeskonnatöös või siis, kui sa kasutad välist pilditöötlusprogrammi) üle salvestatakse, vali UV/pildiredaktoris menüüst Image->Reload (pilt->laadi uuesti) ja sa näed Blenderis uuendatud pilti tema täies hiilguses. Samuti kasuta valikut *Reload* (laadi uuesti) siis, kui oled pildile uusi võre külgi laotanud, et 3D-vaade uuendaks pilte objekti külgedel.

Kui sa muudad pildifaili asukohta, võib Blender seda enam mitte leida ja sa pead selle uult asukohalt ise sisse lugema, valides menüüst Image->Replace (pilt->asenda). Selle valiku abil saad kasutada UV-laotuse jaoks ka täiesti uut pilti.

Uued pildid

Vajutades uue pildi nuppu New Image, esitatakse sulle hulk valikuid. Seda genereeritud pilti on võimalik ka hiljem muuta seadistuste paneelis Properties:

Image Name (pildi nimi)

Määrab genereeritud pildi nime

Width (laius) ja Height (kõrgus)

Määravad pildi suuruse pikslites

Color (värv)

Määrab vaikevärvi, millega täidetakse tühi pilt.

Alpha (läbipaistvus)

Lisab pildile alfakanali (läbipaistvuse)

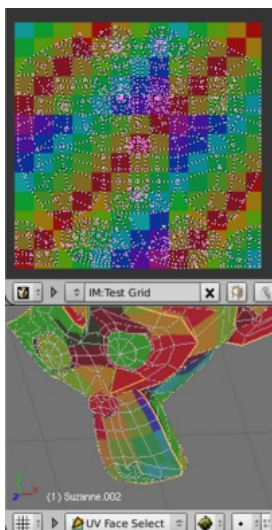
UV Test (UV-test)

Loob UV-testvõrgu pildi, mis võimaldab hinnata UV-laotuse kvaliteeti ja vähendada külgede venitusi. UV-testvõrgul on kaks võimalikku tüüpi, mida saab määrata pärast pildi loomist.

32 bit

Loob 32-bitise pildi. See pilt võtab rohkem kettaruumi, kuid on värvide osas palju täpsem kui tavaline 8-bitine pilt. 32-bitine pilt sobib suures plaanis objektide ja suurte värvüleminekute jaoks.

Testvõrgu kasutamine



UV-testvõrgu abil saab leida tekstuuri lubamatuid venitusi või moonutusi külgedel. Kui tekstuur moodustab pideva ühtlase mustri ja sa tahad, et see tekstuur kataks objekti nagu riie, siis sa kindlasti EI taha, et tekstuur oleks mõnel küljel välja venitatud (välja arvatud juhul, kui see riie peab nägema välja nagu stretšmaterjal).

Renderdamisel määrab testvõrk võre külgede värvi. UV-tekstuuri suurus on määratud genereeritud pildi suurusega. UV-pildi saab salvestada, valides menüüst Image->Save (pilt->salvesta).

Pildi seadistused

Kui pilt on UV-redaktorisse laaditud või uus pilt genereeritud, ilmub seadistuste paneelile Properties uus jaotus. Esimene nuppude rida võimaldab sul:

- Sirvida pilte Blenderi dokumendis (Browse)
- Muuta pildi nime
- Teha pildist võltskasutaja (Fake User)
- Luua uus pilt (New Image)
- Laadida pilt failist (Open)
- Ühendada andmeblokk lahti (Unlink Datablock)

Allika menüüst Source saab valida pildi tüübi. Igal tüübil on erinevad valikud:

Generated (genereeritud)

Loob uue pildi:

Pildi laius (Width) ja kõrgus (Height) pikslites.
Blank (tühi)

Loob tühja pildi

UV grid (UV-testvõrk)

Loob malelauamustri, mille igal ruudul on värviline rist.

Color Grid (värvivõrk)

Loob keerulisema värvilise ruudustiku, millel iga ruudu asukoht on tähistatud tähtede ja numbritega.

File (fail)

Laadib pildi failist:

Fields (poolkaadrid)

Kasuta seda, kui pilt koosneb poolkaadritest. Võimalik on määrata kas ülemine enne (Upper First) või alumine enne (Lower First)

Premultiply (eelkorruta)

Muudab iseseisva alfakanaliga RGB eelkorratud RGB-ks.

Movie (film) ja Sequence (pildijada)

Frames (kaadreid)

Määrab kasutatavate kaadrite arvu

Start (algus)

Filmi/jada alguskaader

Offset (nihe)

Nihuta animatsioonis kasutatava kaadri numbrit

Fields (poolkaadrid)

Määrab, mitu poolkaadrit renderdatud kaader sisaldab (2 poolkaadrit on üks kaader)

Auto Refresh (uuenda automaatselt)

Uuendab pilti alati, kui kaader vahetub.

Cyclic (korduv)

Pärast filmi/jada lõppu alusta piltide näitamist jälle algusest.

Pilditekstuuri muutmine

Sa võid loodud tekstuuri muuta kas

- värvides teda skripti "UV Painter" abil pildi tipuvärvidega,
- eelrenderdades pildile võre väljanägemise (*Render Bake*),
- maalides pilti tekstuurimaalija (*Texture Paint*) abil,
- kasutades välist pilditöötlus- või joonistusprogrammi,
- kasutades viimaste Blenderi versioonide projektsioonmaalimise (*projection painting*) tööriista
- või ülaltoodud meetodite kombinatsiooni abil.

Esimesed kolm valikut, UV-maalija (*UV Painter*), eelrenderdamine (*Render Bake*) ja tekstuuritöötleja (*Texture Baker*) asendavad olemasoleva pildi loodud pildiga. Pildi muutmiseks ja detailide lisamiseks saab kasutada tekstuurimaalijat (*Texture Paint*) ja väliseid programme. Sõltumata sellest, millist meetodit sa kasutad, pead sa viimaks kas

- salvestama pildi eraldi faili (näiteks JPG värvipiltide jaoks, PNG (RGBA režiimis) läbipaistvusega piltide jaoks) või
- pakkima pildi Blenderi faili (*UV/Image Editor Image->Pack as PNG*) (UV/pildiredaktor->paki PNG faili)
- või tegema mõlemat.

Välisesse faili salvestamise eelis on see, et sa saad hõlpsasti tekstuuri vahetada: lihtsalt kopeeri selle asemele teine pilt. Samuti saad sa kasutada sellise pildi töötlemiseks väliseid programme. Pakkimise eelis on see, et kogu sinu projekt on ühes .blend-failis koos ja seega pead sa haldama ainult ühte faili.

Sa saad pildi värvid ümber pöörata, valides pildimenüüst Image ümberpööramise käsu Invert.

Pildi asendamine eelrenderduse tulemusega

[Eelrenderduse \(Render Bake\)](#) abil saab olemasoleva pildi välja vahetada, sõltuvalt renderdatud

- [tipuvärvidest](#) (*Vertex Paint*)
- normaalidest (faktuurist)
- protseduurilistest materjalidest, tekstuuridest ja valgustusest
- hajusvalgusest (*Ambient Occlusion*)

Pildi loomine/muutmine välises programmis

Sa võid luua tekstuuri oma lemmik-pilditöötlusprogrammis, kasutades juhisenä eksportitud UV-laotust. Siis salvesta muudatused ja laadi see pilt Blenderi küljevaliku režiimis menüü *Image->Open* (Pilt->ava) abil võre soovitud (ja aktiivse) UV-kihi tekstuuriks.

Kui kasutad tööriista Edit Externally (muuda välises programmis), mis asub pildimenüüs Image, avab Blender kasutaja seadistustes (User Preferences) määratud pilditöötlusprogrammi ja laadib sellesse muudetava pildi.

Pildi muutmine tekstuurimaalijaga

Vali UV/pildiredaktori menüüst *Image->New*(pilt->uus). Seejärel maali otse võrele, kasutades [tekstuurimaalijat \(Texture Paint\)](#).

Piltide salvestamine

Blenderi pildimenüü (Image) tööriistadega loodud või muudetud pilte võib salvestada välistesse failidesse. Kui pildid juba on failid, kasuta salvestamise käsku Save (Alt+S). Kui pildid on genereeritud või sa tahad salvestada nad uue nimega, võid kasutada ka käsku Save As (F3). Koopia salvestamise käsuga Save as Copy, (F3) salvestatakse küll uue nimega fail, kuid jäetakse pildiredaktoris kehtima vana nimi.

Tekstuuri määramine külgedele

Et külgedele tekstuuri määrata, vali soovitud küljed UV-aknas küljevaliku režiimis. Kui neile ei ole veel tekstuuri määratud, peaks taust olema tühi. Kui külj või küljed on valitud, kas vajuta päise sirvimise ikooni Browse ja vali pilt või lae uus pilt failist.

Kui ainult osadel külgedel on tekstuur määratud, siis ühe määramata tekstuuriga külje lisamisel valikusse muutub taust tühjaks. Kui aga pildi kinnitus (Image Pinning) on sisse lülitatud (vajutades knopka pilti päises), jääb viimati näidatud pilt alles sõltumata valikust.

Aktiivse pildi asendamine

Tuleta meelde, et igal küljega on seotud oma UV-koordinaadid ja viide pildile. Et seostada külj teise pildiga, lihtsalt vali see külj või küljed ja kasuta UV/pildiredaktori pildi menüüs asendamise valikut Replace, mis asendab kehtiva pildi failist (näiteks JPG või PNG) loetud pildiga.

Piltide pakkimine Blenderi faili

Kui sa pakid pildid .blend-faili, pakitakse hetkel kehtivad versioonid kõigist UV-tekstuuridest. Kui sa neid pildifaile hiljem muudad, ei pakita muudetud pilte automaatselt uuesti. Selle asemel kasutatakse vanu versioone. Et pilte uuendada, tuleb nad uuesti pakkida või

uuesti laadida.

Pildi pakkimiseks vali pildimenüüst Image pildi pakkimise valik Pack Image. Lahtipakkimiseks vali samast menüüst Remove Pack (lõpeta pakkimine).

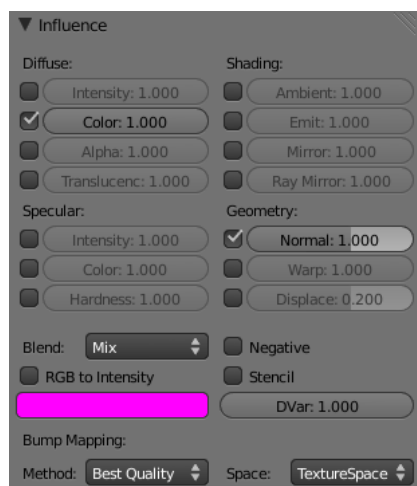
Meetod *File->Append* (fail->lisa juurde) siseneb automaatselt .blend-failidesse ja näitab pilditekstuure, mis neisse on pakitud. Suurepärane piltide allikas on ka vaba (autorikaitsest prii) Blender Texture CD. Lisaks võib tasuta (ja tasulisi) tekstuure leida paljudest teistest kohtadest. Kõik tekstuudid plaadil Elephants Dream CD on vabalt saadaval [CC-BY 2.5](https://creativecommons.org/licenses/by/2.5/) litsentsi alusel.

- [Doc:ET/2.6/Manual/Textures/Influence/Material](#)
- [Doc:ET/2.6/Manual/Textures/Influence/World](#)
- [Doc:ET/2.6/Manual/Textures/Influence/Particles](#)

Materjalitekstuuri mõju

Tekstuurid võivad muuta mitte ainult materjali värvi, vaid ka paljusid teisi omadusi. Seda, milliseid materjali omadusi tekstuur muudab, saab määrata paneelil Influence (mõju).

Pind ja traatmaterjalid



Tekstuuri mõju paneel Influence pindmaterjali (Surface) kasutamisel

Märkus

Tekstuuri valikud pindmaterjali (Surface) ja traatmaterjali (Wire) ning mõnedel juhtudel ka mahulise materjali (Volume) ja halomaterjali (Halo) puhul.

Hajusus (*Diffuse*)

Intensity (intensiivsus)

kui palju tekstuur mõjutab hajususe peegeldust

Color (värv)

Kui palju tekstuur mõjutab materjali värvi (RGB väärtusi)

Alpha (läbipaistvus)

Kui palju tekstuur mõjutab materjali läbipaistvust. Vaata õppetükki [Alfa kasutamine läbipaistvate objektide puhul](#). Lisaks kasuta valgusel ja mitme tekstuurikanali kombineerimisel Z-läbipaistvust (Z Transparency).

Translucency (läbikumamine)

Määrab, kui palju tekstuur mõjutab materjali läbikumavust.

Läige (*Specular*)

Intensity (intensiivsus)

Kui palju tekstuur mõjutab materjali läikepeegelduse intensiivsust

Color (värv)

Mõjutab läike (Specular) värvust, st läikival materjalil lampide poolt tekitatud peegelduste värvi.

Hardness (tugevus)

Mõjutab läigete selgust ja teravust. Väärtus 1 vastab läikivusele 130, väärtus 0.5 läikivusele 65.

Varjutamine (*Shading*)

Ambient (üldvalgus)

Mõjutab materjalile langevat üldvalgust.

Emit (kiirgus)

Mõjutab materjali poolt kiiratava valguse hulka.

Mirror (peegeldus)

Mõjutab materjalilt peegelduvat värvi. See töötab keskkonnapiilide ja kiirtejälitusega peegelduste kasutamisel.

Ray Mirror (kiirtepeegel)

Mõjutab kiirtejälitusega peegelduse tugevust.

Geomeetria (*Geometry*)

Normal (normaal)

Nimetatakse ka pinnafaktuuriks; määrab, kui palju tekstuur mõjutab pinnanormaalide suunda. Seda kasutatakse faktuurikaartidega illusoorsete pinnadefektide, konarlikkuse või reljeefsete pindade loomiseks.

Warp (vääna)

Väänamine võimaldab tekstuuridel muuta/moonutada järgnevate tekstuurikanalite tekstuurikoordinaate. Moonutus kehtib kõigi järgnevate kanalite jaoks, kuni uue väände määramiseni. Kui muuta see väärtus nulliks, siis vääne puudub.

Displace (nihuta)

Mõjutab tippude nihet [nihkekaartide](#) (*Displacement Maps*) kasutamisel.

Segamine (*Blending*)

Blend (sega)

Kuidas selle kanali väärtus suhestub eelnevate kanalite väärtustega. Vaata [komposiitimis-segamissõlme](#) alt täpsemat kirjeldust ja näiteid iga segamismeetodi kohta.

RGB to intensity (RGB intensiivsuseks)

Kui see on sisse lülitatud, kasutatakse RGB teksture (määravad värvi) intensiivsustektuuridena (määravad ühe kanali väärtuse).

Blend Color (segatav värv)

Määrab, mis värv segatakse materjali värvile vastavalt tekstuuri väärtusele juurde, kui intensiivsustekstuur on seostatud värviga. Klõpsa värvivalijal või määra väärtus RGB väljadega.

Negative (negatiivne)

Tekstuuri mõju on ümber pööratud. Tavaliselt tähendab valge "sees" ja must "väljas", Negative (negatiivne) vahetab selle ära.

Stencil (matriits)

Käesolevat tekstuuri kasutatakse maskina järgnevate tekstuuride jaoks. See on praktiline poolläbipaistvate tekstuuride ja "määrumiskaartide" puhul. Must värv muudab vastavad pikslid "mittetekstuuritavaks". Matriits töötab nagu kihi mask 2D-programmis. Matriitsi mõju (blokeeritud piksleid) ei saa edasiste tekstuuridega tühistada, vaid ainult laiendada. Sisendiks tuleb kasutada intensiivsuse kaarti.

DVar

Lisatav väärtus (mitte RGB tekstuuride jaoks). Väärtus, mille intensiivsustekstuur segab olemasoleva kanali väärtusega. Kaks näidet:

- Kiirguse (Emit) väärtus on tavaliselt 0. Kui tekstuur on seostatud kiirgusega, saad sa maksimaalselt tugeva efekti, sest DVar on vaikimisi 1. Kui sa määrad DVar väärtuseks 0, ei ole tekstuuril mingit efekti.
- Kui sa tahad kasutada (osaliselt) läbipaistvat materjali ja seostad tekstuuri läbipaistvusega (Alpha), ei juhtu vaikeparameetrite puhul mitte midagi, sest vaikeläbipaistvuse (Alpha) väärtus materjali paneelil Material on 1. Seega pead sa läbipaistva materjali saamiseks määrama DVar väärtuseks 0 (ja muidugi lülitama sisse ka Z-läbipaistvuse (Z Transparency)). See on algajate jaoks tavaline probleem. Teine meetod: määra materjali läbipaistvuse (Alpha) väärtuseks 0 ja jätta Dvar väärtuseks 1. Loomulikult on tekstuuri läbipaistvus sel juhul ümber pööratud.

Bump Mapping (faktuurikaart)

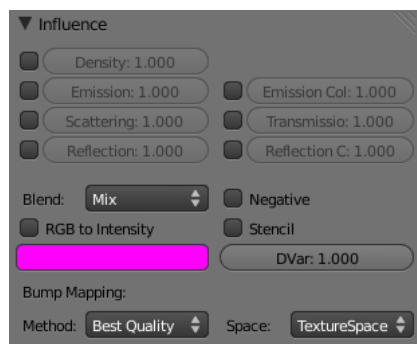
Faktuurikaartide valikud.

Method (meetod)

Best Quality (parim kvaliteet), Default (vaikimisi), Compatible (ühilduv), Original (algne)

Space (koordinaatsüsteem)

Texture Space (tekstuuri), Object Space (objekti), View Space (kaamera)

Mahulised materjalid

Tekstuuri mõjupaneel Influence mahulise materjali (Volume) puhul

Spetsiaalsed valikud mahuliste materjalidele (Volume)

Density (tihedus)

Kui palju tekstuur mõjutab materjali tihedust.

Emission (kiirgus)

Kui palju tekstuur mõjutab mahutekstuuri kiirguse tugevust.

Scattering (hajuvus)

Kui palju tekstuur mõjutab hajuvust.

Reflection (peegeldus)

Kui palju tekstuur mõjutab hajunud valguse heledust

Emission Color (kiirguse värv)

Kui palju tekstuur mõjutab kiirguse värvi.

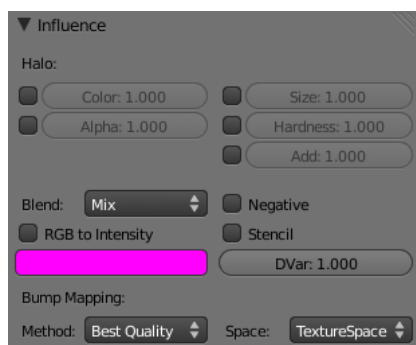
Transmission (läbiv valgus)

Kui palju tekstuur mõjutab läbiva valguse (valgus, mis ei ole hajunud) värvi.

Reflection Color (peegelduse värv)

Kui palju tekstuur mõjutab sellest edasi paiskunud valguse värvi.

Halo materjalid



Tekstuuri mõjupaneel Influence halomaterjali (Halo) puhul

Spetsiaalsed valikud halomaterjalidele (Halo)

Size (suurus)

Kui palju tekstuur mõjutab kiirte peegeldumist.

Hardness (teravus)

Kui palju tekstuur mõjutab teravust.

Add (läbikumavus)

Kui palju tekstuur mõjutab läbikumavust.

Faktuuri- ja normaalikaardid

Kirjeldus

Normaalikaardid (Normal Maps) ja faktuurikaardid (Bump Maps) täidavad mõlemad sama ülesannet: loovad pinnavarjutusel mulje detailse 3D-faktuuriga pinnast, lisades muidu siledale pinnale palju näilisi väikesi nurki ja kaldeid. Kuna see muudab ainult iga üksiku piksli varjutamist, ei heida sellised näivad detailid ise varje ega kata teisi objekte. Kui kaamera on pinna suhtes liiga väikese nurga all, võid märgata, et pind on tegelikult ikkagi sile.

Nii faktuurikaardid (Bump Maps) kui ka normaalikaardid (Normal Maps) muudavad pinnanormaali nurka (suunda, mis on vastavas punktis pinnaga risti). See mõjutab vastava punkti valgustatust. Kuigi termineid normaalikaart (Normal Map) ja faktuurikaart (Bump Map) kasutatakse tihti samas tähenduses, on neil teatud erinevused.

Faktuurikaardid (*Bump maps*)

need on tekstuurid, mis määravad **intensiivsuse**, st pikslite suhtelise kõrguse pinna suhtes. Pikslid näivad olevat nihutatud määratud distantsi võrra pinnanormaali suunas. ("Faktuur" määrab ainult nihke piki külje olemasolevat muutumatut normaalvektorit.) Faktuurina saab kasutada nii haltoonides pilte kui ka RGB-tekstuuride (sealhulgas piltide) intensiivsuse väärtusi.

Normaalikaardid (*Normal maps*)

need on pildid, mis R,G ja B kanalite väärtuste järgi määravad normaali täpse suuna. Normaalikaardid on palju täpsemad, sest kodeerivad mitte lihtsalt piksli kaugust pinnast, vaid tema nihet suvalises suunas suvalisel määral. Normaalikaartide puudus on see, et erinevalt faktuurikaartidest, mida on lihtne käsitsi maalida, tuleb normaalikaardid tavaliselt genereerida programmiselt. Tihti kasutatakse selleks modelleeritava objekti suurema detailsusega versiooni.

Normaalikaardid Blenderis kodeerivad normaale järgnevalt:

- Punane kanal (0-255) kodeerib X-väärtusi (-1.0 - 1.0)
- Roheline kanal (0-255) kodeerib Y-väärtusi (-1.0 - 1.0)
- Sinine kanal (0-255) kodeerib Z-väärtusi (0.0 - 1.0)

Kuna normaalid on alati suunatud vaataja poole, pole negatiivseid Z-väärtusi vaja salvestada (nad oleksid niikuinii nähtamatud). Blenderis kasutatakse kogu sinise vahemikku. Osades teistes rakendustes kodeerivad sinise väärtused (128–255) Z-väärtusi (0.0–1.0). Viimatitoodud meetodit kasutab näiteks mäng "Doom 3".

Töö käik

Faktuuri- ja tekstuurikaartide kasutamiseks tuleb teha järgmised sammud:

1. Modelleerida suure detailsusega (*hi-poly*) objekt
2. Eelrenderdada faktuuri- ja/või normaalikaardid
3. Modelleerida madala detailsusega (*low-poly*) objekt
4. Laotada kaart madala detailsusega mudelile, kasutades ühist koordinaadistikku

Vaata modelleerimise peatükist, kuidas võretöötlusvahenditega luua suure detailsusega objekti. Kui palju detaile sa lisad, on täiesti su enda teha. Mida rohkem vagusid, kühme ja muid detaile (nuppe, õmblusi) sa lisad, seda detailsem on ka faktuurikaart.

Lihtsalt öeldes tähendab kaardi eelrenderdamine suure täpsusega võre detailide ülekandmist teisele samasele objektile. See samane objekt peab olema muidu identne suure täpsusega võrega, v.a see, et tal on vähem tippe. Normaalikaardi tegemiseks kasuta Blenderi [eelrenderdust \(Render Bake\)](#).

Madala detailsusega modelleerimine Blenderi võretöötlusvahenditega. Üldreeglina peavad madala detailsusega võrel olema samad või samased küljed. Näiteks võib kõrglahutusega mudeli kõrv koosneda 1000-st küljest. Madala lahutusega mudelil võib need asendada ühe küljega, mis on paigutatud samas suunas nagu suure lahutusega kõrv. (*Vihje*: Selleks võib olla praktiline kasutada Blenderi [paralleelvõre \(Multi Resolution\)](#) töötlejat.)

Normaalikaardi laotamine on tekstuuri lisamine madala detailsusega võrele. Täpsemad juhised, kuidas võre materjalile tekstuure lisada, leiad [tekstuuri-laotuse peatükist](#). Faktuuri- ja normaalikaartide kasutamisel pea silmas:

- Faktuurikaartide kasutamisel seosta tekstuur normaaliga ja lülita sisse No RGB (mitte RGB).
- Normaalikaardi kasutamisel seosta tekstuurinormaaliga.

Kahe objekti koordinaatsüsteemid peavad olema identsed. Näiteks kui sa eelrenderdad, kasutades kõrglahutusega mudeli UV-laotust, pead sa madala lahutusega mudeli UV-laotuse ühitama kõrglahutuse UV-kaardi piirjoontega (vaata [UV-lahtilõike](#) peatükist, kuidas ühitada UV-koordinaate kõrglahutusega laotuse servadega).

Nihkekaardid

Kirjeldus

Nihkekaardid (*Displacement Maps*) võimaldavad renderdatava geomeetria tippe nihutada vastavalt tekstuuri väärtustele. Erinevalt [normaali- ja faktuurikaartidest](#), mis loovad mulje detailidest, muutes pinna varjutust (nagu kirjeldatud eelmises peatükis), tekitavad nihkekaardid võrele reaalsed kühmud, vaod, vallid jne. Sellised võredeformatsioonid heidavad varje, varjavad teisi objekte ja teevad kõike muud, mida võre geomeetria analoogilised muudatused teeksid. Teiselt poolt nõuavad nihkekaardid palju suurema tippude arvuga võret.

Valikud

Nihke suurust saab määrata [mõjupaneeli](#) *Influence* väljade Displace (nihe) ja Normal (normaal) abil.

- Kui tekstuur kodeerib ainult normaale (näiteks Stucci (krohvi)), liigutatakse tippe tekstuuri normaalikanaliga määratud suunas. Väli Normal (normaal) määrab piki normaale nihutamise määra.
- Kui tekstuur sisaldab ainult intensiivsusi (näiteks värvusest tuletatud Magic), liigutatakse tippe piki nende normaale (tipul endal küll ei ole normaali, selle asemel kasutatakse kõrvalasuvate külgede normaalide keskmist). Valged pikslid liigutavad tippu piki normaali väljapoole, mustad sissepoole. Nihke suuruse määrab väli Displace (nihe).

Need kaks režiimi ei välista vastastikku teineteist. Paljud tekstuuritüübid sisaldavad mõlemat liiki andmeid (Clouds (pilved), Wood (pui), Marble (marmor), Image (pilt)). Kummagi tüübi mõju saab vastavate seadete abil eraldi määrata. Nihe intensiivsuse järgi tekitab siledama ja ühtlasema pinna, sest tippe nihutatakse ainult normaali suunas. Nihe normaali järgi tekitab ebaühtlasema pinna, sest tippe nihutatakse paljudes suundades.

Nihke suurus on proportsionaalne objekti mõõtkavaga, mitte tema absoluutse suurusega. See tähendab, et kui sa suurendad objekti objektirežiimis kaks korda, suureneb ka nihe kaks korda, nii et suhteline nihe jääb samaks. Kui sa suurendad objekti muutmisrežiimis Edit, siis nihke absoluutne suurus ei muutu ja seega suhteline nihe väheneb.

Näpunäited

Nihkekaardid liigutavad renderdatud külgi, mitte võre tegelikke külgi. Seega võib pind 3D-vaates näida sile, kuid renderdusel konarlik. Et saada detailirohket pinda, peab võrel olema piisavalt külgi, seega peavad küljed olema väikesed. Kõrge kvaliteediline renderdus nõuab selle meetodi puhul oluliselt rohkem mälu ja protsessoriaega.

Parimast halvamani töötab pinnanihe järgmiste objektitüüpidega. Iga tüübi järel on toodud meetodid, mis mõjutavad külgede suurust renderdamisel:

[Tükeldatud pinnaga](#) võred

Külgede suuruse määrab pinnatükelduse tase. Pinna nihutamiseks on head ühtlased normaaliid (sile pind).

Käsi (muutmisrežiimis Edit Mode) [tükeldataud](#) võred

külgede suuruse määrab tükelduste arv. (Võib kombineerida ka eelmise meetodiga.) Antud meetodi puhul nihutatakse renderdusel täpselt samasugust tükeldatud pinda kui eelmise puhul, kuid objekti muutmine on aeglasem, sest OpenGL peab joonistama kõik võre tegelikud küljed. (Sa ei saa tükeldatud külgede näitamist muutmisrežiimis välja lülitada.)

[Metaobjektid](#)

külgede suuruse määrab renderdusvõrgu tihedus. Tihe võrk == rohkem külgi.

Alltoodud objekte saab kasutada, kuid hetkel ei tööta nad hästi. Soovitav on seda tüüpi objektid enne renderdamist võreks konverteerida.

Avatud [NURBS pinnad](#)

külgede suuruse määrab U/V-pinnalahutus (Surface Resolution). Suurem number tekitab rohkem külgi. (Pane tähele normaalide vigu.)

Suletud [NURBS-pinnad](#)

külgede arvu määravad pinnalahutuse (Surface Resolution) valikud. (Pane tähele normaalide vigu ja seda, kuidas õmblus on näha.)

[Kõverad ja tekst](#)

külgede arvu määravad pinnalahutuse (Surface Resolution) valikud. Suurem number tekitab rohkem külgi. (Pane tähele, et suurtel tasastel pindadel on vähe külgi, mida nihutada.)

Nihutustöötaja (*Displace Modifier*)

Kui sa tahad pinna nihutamist täpsemalt juhtida, siis tuleb sul tõenäoliselt kasutada [nihutustöötajat](#). Sellel meetodil on palju erinevaid valikuid, mille abil saad nihet täpselt soovi järgi sättida.

Maailm (World)



Maaailma paneel (World)

Blenderisse on ehitatud mitu väga huvitavat sätet, millega saad oma renderdust ilusa tausta ning huvitavate 'sügavuse' efektide lisamisega täiustada. Neid sätteid saab muuta maailma (World) kontekstimenüüst. Vaikimisi on maailm suhteliselt lihtne ja üheülbane. Saad seda maailma muuta või lisada täiesti uue.

Sul on järgmised valikud:

[Taust](#)

Maaailma tausta värv ja tekstuur koos spetsiaalsete laotuse koordinaatide sätetega.

[Udu](#)

Lisa oma stseenile udu, mis suurendab näilist sügavust.

[Tähed](#)

Katab tausta juhuslike helendavate punktidega.

Kuigi nende maailma sätete kaudu saad oma stseenile väikese vaevaga efekte lisada, eelistatakse sageli [komposiitmissõlmi](#), sest kuigi neid on keerulisem selgeks õppida, pakuvad nad suuremat kontrolli ja lisavalikuid. Näiteks võib stseeni sügavust ja ruumilist selgust veelgi enam suurendada Z-väärtuse (kaugus kaamerast) või normaalide (pindade suuna) filtreerimine.

Märkus

Mõned sätted Maaailma paneelis mõjutavad Blenderis valgustust ning seetõttu leiad nad [valgustuse](#) peatükist (vaata [üldvalgust](#) ([Ambient Light](#)), [säri](#) ([Exposure](#)) ja [kaudvarju](#) ([Ambient Occlusion](#))). Kasutades päikese valgustit (Sun Lamp), on valgusti menüüs Lamp võimalik muuta taeva ja atmosfääri sätteid (Sky & Atmosphere).

Maaailma tagaplaan

Kirjeldus

Maaailma nupud lubavad sul oma stseeni varjutuse üldpõhimõtteid määrata. Siit saad muuta keskkonna värvi ning lisada eriefekte nagu udu, kuid üks põhilisi maaailma (World) kasutamisi on taustavärv määramine.

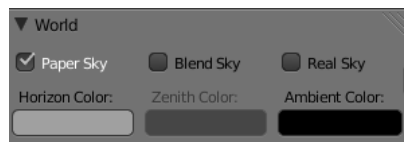
Taustapildi kasutamine renderdamisel

Taustapildi kasutamiseks renderduse taustana vaata [väljundi paneelil \(Output Panel\)](#) olevat taustapildi välja (*BackBuf image*)

Taustapildi kasutamine 3Ds

Taustapildi kasutamiseks 3D-vaates (näiteks modelleerimise näidisenä) vaata [taustapildi kasutamist](#)

Seaded



Maaailma paneel (World)

Horizon Color (Horisondi värv)

RGB-süsteemis värv horisondil

Zenith Color (Seniidi värv)

RGB-süsteemis värv seniidis (pea kohal)

Valitud värvide tõlgendus oleneb sellest, milline taeva (Sky) tüüp on valitud.

Pole valitud

Kui pole valitud ühtegi neist kolmest nupust, on taevas lihtsalt ühevärviline (kasutab horisondi värvi).

Paber taevas (Paper Sky)

Selle valikuga säilib värvüleminek oma omadused, kuid on pildil lõigatud (see jääb horisontaalsele tasapinnale (x-y tasapinnaga paralleelseks): ükskõik milline kaamera nurk ka poleks, on horisont alati pildi keskel).

Sulandatud taevas (Blend Sky)

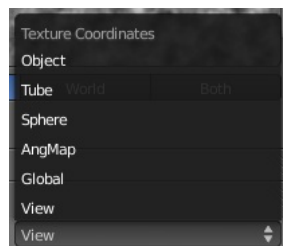
Taustavärv sulandub horisondilt seniidini. Kui ainult see nupp on sisse lülitatud, jookseb värvüleminek renderdatud pildi alumisest servast ülemise servani, hoolimata kaamera vaatenurgast.

Päris taevas (*Real Sky*)

Selle valikuga on tulemuseks kaks sulatust, nadiirist (seniidiga sama värvi) horisondini ja horisondist seniidini; sulandumine sõltub kaamera vaatenurgast ning on seetõttu ka realistlikum. Horisondi värvi kasutatakse täpselt horisondil (x-y tasapinnal) ning seniidi värvi kasutatakse kahes punktis, mis asuvad vertikaalselt (z-teljel) kaamera kohal ja all.

Tekstuurid

Ühe värvi (või kahe värvi sulatuse asemel) võib Blender kasutada ka kahemõõtmelist pilti, mis laotatakse väga suurele kuubile või kerale, mis ümbritseb tervet stseeni, või mis laotatakse stseeni ümbritsevale virtuaalsele ruumile.



Tekstuurikoordinaatide

(Texture Coordinates)

hüpikmenüü

Maaailma teksture saab näha tekstuuride menüüs (vali esmalt maailm (World) ja seejärel (Texture)). Neid kasutatakse analoogselt materjali tekstuurile, kuid on ka mõningad erinevused. Teksture saab laotada:

Vaate järgi (*View*)

Vaikeorientatsioon, mis joondab teksturi lõpliku renderduse koordinaatidega

Globaalselt (*Global*)

Kasutab globaalkoordinaate

Nurklaotus (*AngMap*)

Sellega saad pildi stseeni ümber paiknevale kuplile laotada. Seda saad kasutada pildipõhiseks valgustamiseks, kui kaudvari (Ambient Occlusion) on seatud taeva värvile. Tavaliselt vajad sa selliseks nurklaotuseks kõrge dünaamilise diapasoniga pilti (*high dynamic range image, HDR*), mis näeb välja nagu imelik kerakujuline pilt.

Kera (*Sphere*)

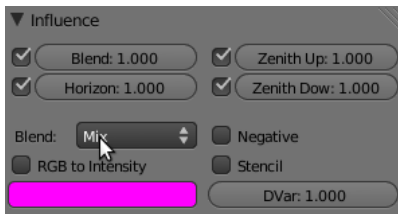
Kera laotus, samaneb materjalide omaga

Toru (*Tube*)

Mähib nelinurkse teksturi silinderjalt ümber, samanedes materjalide omaga

Objektipõhine (*Object*)

Asetab teksturi valitud objekti kohalikust tekstuuriruumist sõltuvalt



Tekstuuri mõju paneel (Influence)

Tekstuur mõjutab ainult värve, kuid teeb seda neljal erineval moel:

Sulandatud (*Blend*)

Horisondi värv ilmub seal, kus tekstuur erineb nullist

Horisont (*Horizon*)

Mõjutab horisondi värvi

Ülemine seniit (*Zenith Up*)

Mõjutab pea kohal oleva seniidi värvi

Alumine seniit (*Zenith Down*)

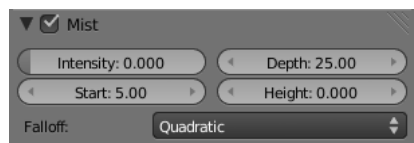
Mõjutab jalge all oleva seniidi (nadiiri) värvi

Udu (*Mist*)

Kirjeldus

Udu võib sinu renderduse sügavuse illusiooni oluliselt suurendada. Udu tekitamiseks muudab Blender kaugemal olevad objektid läbipaistvamaks (vähendab nende alfaväärtust), nii et nende objektide värvus sulandub enam tausta värvusega. Kui udu on sisse lülitatud, on kaamerast kaugemal olevate objektide alfaväärtus üha väiksem.

Seaded



Udu paneel (*Mist*)

Udu märkeruut (*Mist*)

Lülitab udu sisse ja välja

Intensiivsus (*Intensity*)

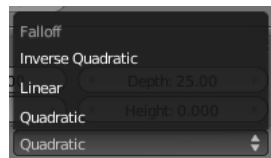
Üldine udu minimaalne intensiivsus ehk tugevus.

Algus (*Start*)

Lähim distantis kaamerast, millelt udu mõju avaldama hakkab

Sügavus (*Depth*)

Kaugus udu alguse punktist Start, millest kaugemale ei näe Objektid, mis on kaamerast kaugemal kui väljade Start+Depth koguväärtus, on udu poolt täielikult peidetud.



Udu hajuvuse hüppikmenüü

Falloff

Kõrgus (*Height*)

Reaalsema efekti saavutamiseks paneb udu intensiivsuse kõrgusega vähenema. Kui välja väärtus on suurem kui 0, määrab see Blenderi ühikutes vahemaa koordinaadist z=0, mille jooksul udu jõuab maksimaalsest intensiivsusest (all) nullini (ülal).

Hajusus (Falloff)

Udu langussamm (kvadraatne (Quadratic)/lineaarne (Linear)/pöördkvadraatne (Inverse Quadratic)). Need seaded kontrollivad udu tugevuse muutuse kiirust, mida kaugemale see kaamerast jääb.

Udu kaugus

Udu kauguse näitamiseks 3D-vaates vali oma kaamera, mine kaamera menüüsse ja lülita sisse udu valik Show Mist.

Kaamera näitab nüüd udu piire kaamerast välja jooksva joonena, mis algab alguspunktis Start ja on sügavuse väärtusega Depth võrdses pikkuses.

Udu (*Mist*) visualiseerimise paremaks hindamiseks vajuta ⇧ ShiftNum1 samal ajal, kui kaamera on valitud (perspektiivvaate sisse ja välja lülitamiseks vajuta Num5). See asetab 3D-vaate otse kaamera kohale ja suunab selle nurga alla.

Läbipaistvus

Kuna udu töötab läbipaistvuse muutmisega, võib sellega mõnikord kaasneda objektide osaline läbipaistvus, kui nad seda olla ei tohiks. Üks möödapääs sellest on udu sätete soovikohane määramine, kuid seejärel udu maha keeramine. Udu andmed on komposiitimiseks ikka veel saadaval, kuigi udu ise on välja lülitatud. Kasuta [komposiitide tegemist \(Do Composite\)](#) ja [sõlmeredaktori](#), et saata udu käik [alfa sõlme \(AlphaOver\)](#) ning sulandada taustavärv (või renderduskiht otse taevaga) kokkurenderdatud pildiga. See tekitab udu, kuid kuna udu ise on välja lülitatud, jääb objekti läbipaistvus (või selle puudumine) alles.

Näide



Udu näide

Selles näites ([.blend-fail](#)) on udu kõrguse sätet (Height) piiratud, mis tekitab pörandat katva suitsu efekti.

Selle lihtsa stseeni inspiratsiooniks oli [Stefan Morelli pilt Arc Sci-Fi Corridor](#).

Tähed (*Stars*)

Kirjeldus

Tähed on juhuslikult asetatud halotaolised objektid, mis taustale ilmuvad.

Sätted



Tähe paneel (Star)

Suurus (*Size*)

Tähe halo reaalne suurus. Parem on hoida selle väärtust väiksemana kui pakutav vaikeväärtus. Pikslist väiksem materjal tekitab imepisikesed täpid. Sellisena on tulemus oluliselt realistlikum.

Värvid (*Colors*)

Lisab muidu valgetele tähtedele värvitoone.

Minimaalne kaugus (*Min. Dist*)

Minimaalne kaugus kaamerast, kuhu tähti asetatakse. See peaks olema suurem kui sinu stseeni kaamera ja *kaugeima* objekti vaheline kaugus, kui sa ei soovi oma objektide ette ilmuvate tähtedega riskida.

Vahekaugus (*Separation*)

Tähtedevaheline *keskmine* kaugus. Tähed on kolmemõõtmelised ning seega paigutatakse nad ruumi ning mitte pildi peale.

Sissejuhatus

Pärast oma tegelase lõpetamist on sul vaja meetodit, kuidas teda animeerimisel või poseerimisel liigutada – siin tulebki mängu taageldamine.

Taageldamine on tegelase võrele skeleti lisamine, nii et teda saab vajalikul viisil deformeerida ja poseerida. Sellised meetodid ja muuta otseselt võret ning neid muutusi saab lihtsalt asendada, tühistada ja teiste poosidega kombineerida.

Skeletid (*Armatures*)

Skelett on taageldamiseks kasutatav objektitüüp, mille paljud omadused on laenatud tegelikelt skelettidelt.

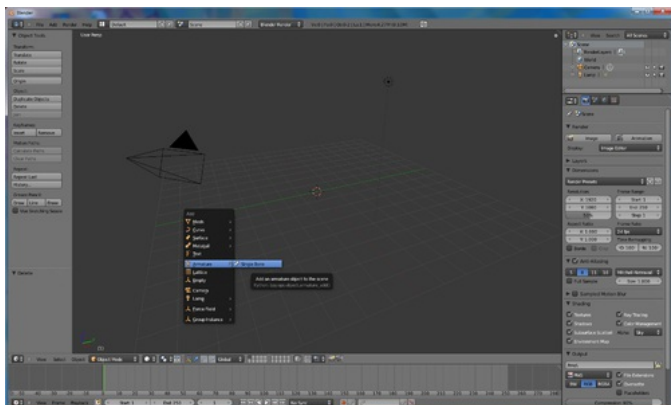
Sinu esimene skelett

Et edasistest jutust aru saada, proovime Blenderis lisada ühe vaikeskeleti.

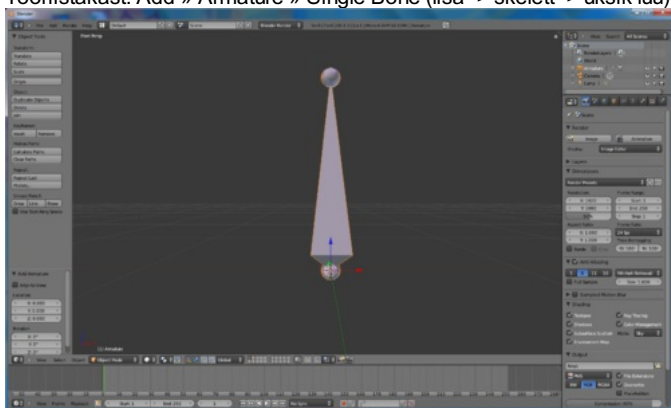
(Pane tähele, et muutmist kirjeldatakse detailsemalt [skeleti muutmise peatükis](#)).

Ava vaikestseen ja seejärel:

- kustuta stseenist kõik objektid
- veendu, et kursor oleks koordinaatide keskpunktis ⇧ ShiftC
- vajuta 1 NumPad, et näha maailma eestvaates
- seejärel, kas:
 - vali peamenüüst Add > Armature > Single Bone (lisa -> skelett -> üksik luu)
 - - või - olles 3D-vaates, lisa skelett, valides ⇧ ShiftA » Armature » Single Bone (skelett -> üksik luu)
- vajuta ⌘ Home, et näha skeetti maksimumsuurendusega



Tööriistakast: Add » Armature » Single Bone (lisa -> skelett -> üksik luu)



Vaikeskelett

Skeletiobjekt

Nagu sa näed, on skelett samasugune nagu kõik teised Blenderi objektid:

- Tal on keskpunkt, asukoht, pööre ja mõõtkava faktor.
- Tal on objektiandmete andmeblokk (*ObData*), mida saab muutmisrežiimis Edit muuta.
- Teda saab linkida teistesse stseenidesse ja samu skeletiandmeid saab kasutada mitme objekti jaoks.
- Kõik animatsioonid, mida sa teed objektirežiimis Object, muudavad ainult tervet objekti, mitte skeleti üksikuid luid (viimaste ühekaupa animeerimiseks kasuta poseerimisrežiimi Pose).

Kuna skelett on mõeldud poseerimiseks kas siis staatilises või animeeritud stseenis, on tal üks spetsiaalne asend, mida nimetatakse "puhkeasendiks" (*rest position*). See on skeleti vaikimisi "kuju", kõigi tema luude vaikimisi asend/pööre ja mõõtkava, nagu see on määratud muutmisrežiimis Edit.

Muutmisrežiimis Edit näed skeetti alati puhkeasendis, samas kui objektirežiimis Object ja poseerimisrežiimis Pose töötad sa tavaliselt skeleti "hetkepositsiooniga" (välja arvatud juhul, kui sa lülitad skeleti paneelil Armature sisse puhkeasendi nupu Rest Position).

Skeletipeatüki ülevaade

Peatükis "Skelett" käsitleme me ainult skeetti ennast, täpsemalt järgmisi teemasid:

- Skeleti [paneelid](#)
- [Luude](#) põhialused
- Skeleti erinevad [esitusviisid](#)
- Skeleti [struktuuri tüübid](#)

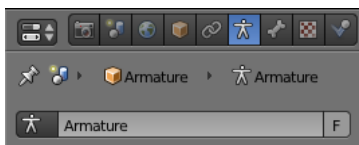
- Kuidas skeleti [osi valida](#)
- Kuidas [skeletti muuta](#)
- Kuidas [muuta luid](#)
- Kuidas [muuta luude omadusi](#)
- Kuidas visandada skelette tööriistaga [Etch-a-Ton](#)
- Kuidas kasutada [skeletimalle](#)

Skeleti paneelide ülevaade

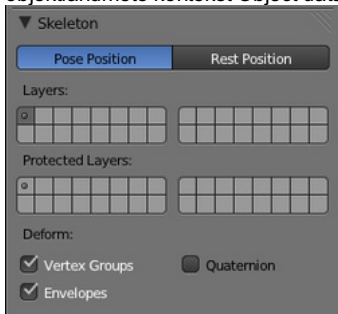
Mode: Objektirežiim Object, muutmisrežiim Edit ja poseerimisrežiim Pose

Panel: Kõik asuvad omaduste akna Properties objektandmete kontekstis Object data

Anname kõigepealt kiire ülevaate erinevatest paneelidest, mis tegelevad skeleti sätetega ja asuvad omaduste akna Properties objektandmete kontekstis Object data:



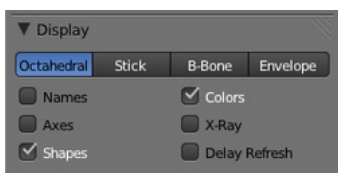
Omaduste akna Properties objektandmete kontekst Object data



Skeleti paneel Skeleton

Skeleti paneel Skeleton (kõik režiimid)

Selles paneelis saad luude komplekte kihtidele jaotada, et nendega töötamist lihtsustada.

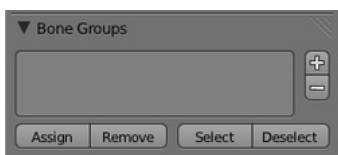


Esituspaneel Display.

Esituspaneel Display (kõik režiimid)

Siin saad määrata ühe neljast viisist, kuidas luid 3D-aknas näidatakse.

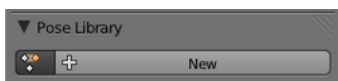
Lisaks on siin veel täiendavaid valikuid, mida kirjeldame lähemalt edaspidi.



Luugruppide paneel Bone Groups

Luugruppide paneel Bone Groups (poseerimisrežiim)

Siin saad paigutada luud gruppidesse, et lihtsustada nende käsitlemist ja haldamist.



Pooside teegi paneel Pose Library

Pooside teegi paneel Pose Library (poseerimisrežiim)

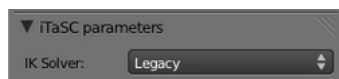
Võimaldab poose hilisemaks kasutamiseks salvestada.



Kummituse paneel Ghost

Kummituse paneel Ghost (kõik režiimid)

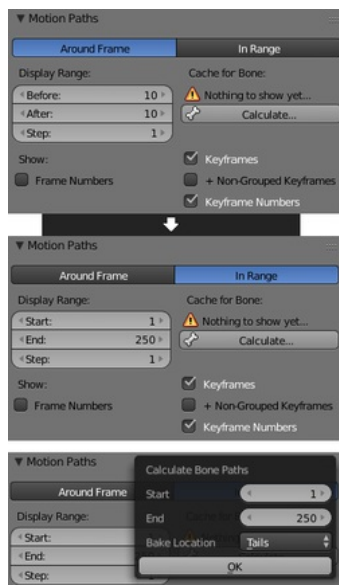
Võimaldab sul näha korraga mitut üksteisele järgnevat poosi, mis on väga kasulik animeerimisel.



iTaSC parameetrite paneel iTaSC parameters

iTaSC parameetrite paneel iTaSC parameters (kõik režiimid)

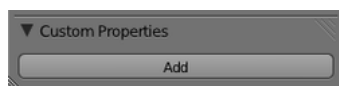
Määrab animatsioonis kasutatava IK lahendaja tüübi.



Liikumisradade paneel Motion Paths

Liikumisradade paneel Motion Paths (poseerimisrežiim)

Sellelt paneelilt saab sisse lülitada sinu skeleti poolt animatsioonis läbitud liikumisraja näitamise.



Täiendavate omaduste paneel Custom Properties

Täiendavate omaduste paneel Custom Properties (kõik režiimid)

Sellel paneelil saab lisada suvalisi parameetreid; kasutatakse skriptimisel.

Luu paneelide ülevaade

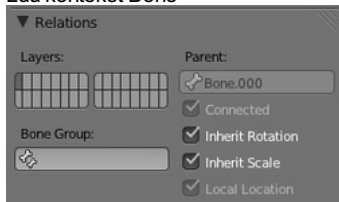
Mode: Objektirežiim Object, muutmisrežiim Edit ja poseerimisrežiim Pose

Panel: Kõik asuvad omaduste akna Properties luu kontekstis Bone

Anname kõigepealt kiire ülevaate erinevatest paneelidest, mis tegelevad luu sätetega ja asuvad omaduste akna Properties luu kontekstis Bone:



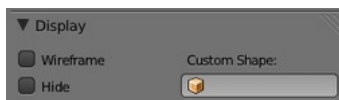
Luu kontekst Bone



Suhete paneel Relations

Suhete paneel Relations (muutmisrežiim)

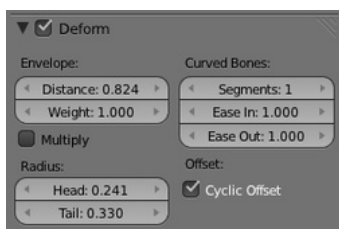
Selles paneelis saad luude komplekte kihtidele jaotada, et nendega töötamist lihtsustada.



Esituspaneel Display

Esituspaneel Display (objektirežiim)

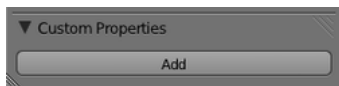
Esituspaneelil saad anda luudele mõne teise juba olemasoleva objekti kuju.



Moontuste paneel Deform

Moontuste paneel Deform (kõik režiimid)

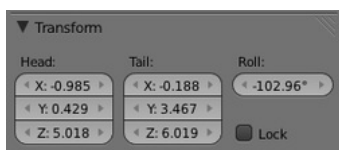
Sellel paneelil saad sa määrata luude põhilisi omadusi.



Täiendavate omaduste paneel
Custom Properties

Täiendavate omaduste paneel Custom Properties (kõik režiimid)

Sellel paneelil saab lisada suvalisi parameetreid; kasutatakse skriptimisel.

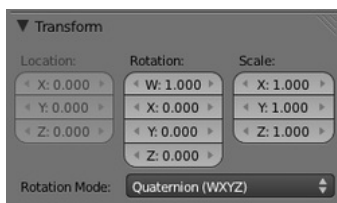


Teisenduste paneel Transform
(muutmisrežiim)

Teisenduste paneel Transform (muutmis- ja poseerimisrežiim)

Muutmisrežiimis võimaldab see paneel määrata üksikute luude asendit ja pööret.

Poseerimisrežiimis saad sa asendit muuta ainult peamisel luul, teistel luudel aga pööret ja skaalat.



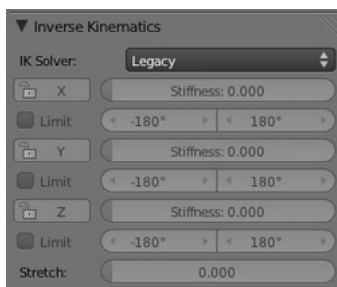
Teisenduste paneel Transform
(poseerimisrežiim)



Teisenduste lukustamise paneel
Transform Locks

Teisenduste lukustamise paneel Transform Locks (poseerimisrežiim)

See paneel on näha ainult poseerimisrežiimis ja võimaldab piirata valitud luude asendi, pöörde ja mõõtkava teisendamist üksikute telgede kaupa.

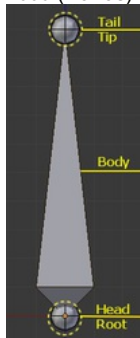


Pöördkinemaatika paneel Inverse
Kinematics

Pöördkinemaatika paneel Inverse Kinematics (poseerimisrežiim)

See paneel võimaldab määrata, kuidas luu või luude grupp pöördkinemaatilises (IK) ahelas käitub.

Luid (*Bones*)



Luu elemendid

Luid on skelettide põhielemendid.

Neil on kolm osa:

- “alguspunkt”, mida nimetatakse **juureks** (*root*) või **peaks** (*head*),
- luu “keha” ise,
- “lõpp-punkt”, mida nimetatakse **tipuks** (*tip*) või **sabaks** (*tail*).

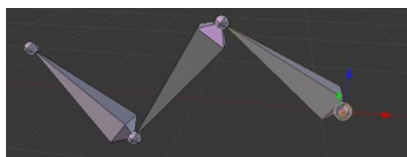
Vali [vaikeskelett](#) ja sisene kiirklahviga \leftrightarrow Tab muutmisrežiimi Edit. Nagu sa näed, saad sa selles režiimis valida luu juurt ja tippu ning neid liigutada, nagu sa teed võre tippudega (ära siiski sellega siin liiga palju aega raiska – luude valimise ja muutmise kohta on olemas eraldi peatükid).

Juur ja tipp (“otspunktid”) määravad oma asukohtadega luu kuju.

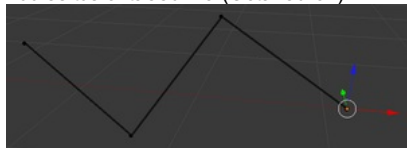
Neil on ka muudetav raadius, millel on küll kasulik ainult ümbrikega deformeerimise puhul (loe altpoolt).

Luude visualiseerimine

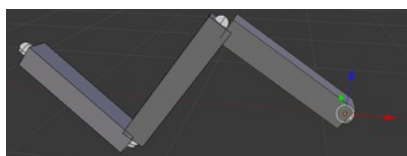
Luid saab visualiseerida erinevate kehadena (oktaeeder (*Octahedron*), pulk (*Stick*), B-luu (*B-Bone*) ja ümbrik (*Envelope*)), muuhulgas isegi sinu enda poolt defineeritud kujuga kehana!



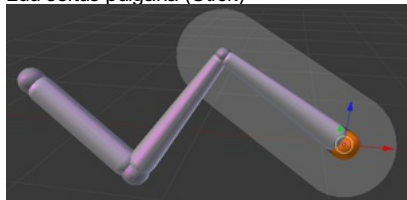
Luu esitus oktaeedrina (*Octahedron*)



Luu esitus pulgana (*Stick*)



Luu esitus pulgana (*Stick*)



Luu esitus ümbrikuna (*Envelope*)

Kuna skeletid koosnevad luudest, räägime me sellest lähemalt peatükis [Skeleti visualiseerimine](#).

Kui telgede nupp Axes skeletipaneelil Armature on sisse lülitatud, näidatakse luu tipus lokaalset koordinaatsüsteemi. *Y-telg on alati suunatud piki luud, juurest tipuni*. Seega on see luu *pöörlemistelg*.



Luu kontekst (*Bone*)

Luude omadused

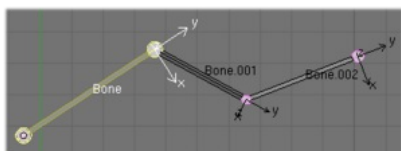
Kui luu on valitud (seega siis kas muutmisrežiimis Edit või poseerimisrežiimis Pose), näidatakse tema omadusi omaduste akna Properties luu nupu kontekstis Bone.

Seal on mitmeid paneele valitud luude omaduste muutmiseks ja paneelide arv võib muutuda, sõltuvalt hetke töörežiimist.

Luude jäikus

Ehkki luud on iseenesest jäigad (st käituvad pulkadena), koosnevad nad üksikutest "segmentidest" ehk väikestest jäikadest tükkidest, mis võivad üksteise suhtes olla pöördes. Vaikimisi on igal uuel luul ainult üks segment ja seega ei saa ta "painduda" piki kõverat: ta on jäik luu.

Sa võid neid segmente näha objektirežiimis Object ja poseerimisrežiimis Pose, juhul kui luu visualiseerimise viisiks on valitud B-bones. Muutmisrežiimis Edit näidatakse luid alati jäikade pulkadena. Pane tähele erijuhtu, et üksikute luude korral ei näe sa neid segmente ka objektirežiimis, sest nad on omavahel joondatud.



B-luudega (B-Bone) skelett muutmisrežiimis



Ahelale on lisatud Bézier' kõver, mille juhtpunktid on luude otstes.



Sama skelett objektirežiimis

Kui sa ühendasid luud [ahelaks](#), arvutab Blender välja Bézier' kõvera, mis läbib kõiki luutippe ning ahelas paiknevate luude segmentid painduvad vastavalt sellele nähtamatule kõverale.

Sa ei saa seda kõverat otseselt muuta, vaid ainult luude omadusi muutes teataval määral juhtida, nagu selgitatakse [muutmise leheküljel](#).

Pildil *B-luudega (B-Bone) skelett muutmisrežiimis* ühendasime me ahelaks 3 luud, millest igaüks koosneb 5 segmentidest. Need on B-luud, kuid nagu sa näed, näidatakse neid muutmisrežiimis Edit jäikade elementidena. Vaata pilti *Sama skelett objektirežiimis*: olles nüüd objektirežiimis Object, näeme, kuidas luude segmentid sujuvalt üksteisega isegi pöördes olles "sulanduvad".

Loomulikult deformeeritakse ka selle ahela poolt mõjutatavat võret sujuvalt vastavalt Bézier' kõverale! Reaalses töös on sujuvalt painduvad luud lihtne võimalus asendada pikki pöördkinemaatika (*Inverse Kinematics*, IK) abil juhitavaid jäikade luude ahelaid...

Siiski – kui ahel mõjutab objekte ja mitte geomeetriat, ei arvestata üksikute segmentide asendit (detailidest räägitakse lähemalt peatükis [nahaga katmine](#)).

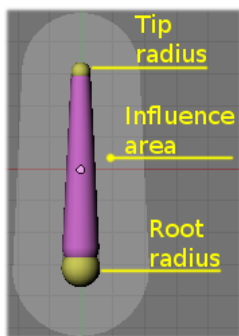
Kui luude esitusviisiks ei ole B-luud (B-Bones), näidatakse luid alati jäikade pulkadena, *kuigi segmentid on neis ikkagi olemas ja funktsioneerivad* (vaata peatükki [nahaga katmine \(ObjData\)](#)).

See tähendab, et isegi kui luude esitusviis on näiteks oktaeedriline (Octahedron), aga mõned luud ahelas koosnevad mitmest segmentidest, deformeerivad nad geomeetriat sujuvalt...

Luude mõju

Põhiliselt juhivad luud geomeetriat, pannes tipud luule "järgnema". Selle jaoks on vaja määrata, **kui palju** luu teatud tippu mõjutab.

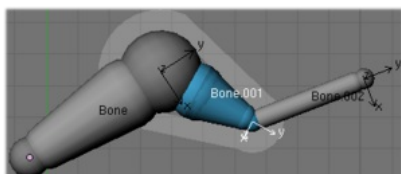
Kõige lihtsam viis on panna iga luu mõjutama seda osa geomeetriast, mis on temast teatavas kaugusevahemikus. Seda kutsutakse *ümbrikumeetodiks*, sest sel juhul mõjutab iga luu ainult seda osa geomeetriast, mis asub luud ümbritseva "ümbriku" (*envelope*) ehk mõjuala sees.



Luu ümbriku esitusviisiga (Envelope) muutmisrežimis Edit

Kui luu esitusviisiks on ümbrik (Envelope), siis sa võid muutmis- ja poseerimisrežimis näha tema ruumilist mõjuala, mis sõltub:

- Kauguse sätte Distance väärtusest
- Juure ja tipu raadiustest.

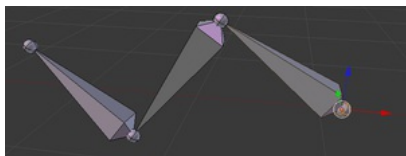


Meie skelett ümbriku esitusviisiga (Envelope) poseerimisrežimis.

Kõiki neid mõju seadeid käsitletakse lähemalt [nahaga katmise peatükis](#).

Skeleti visualiseerimine

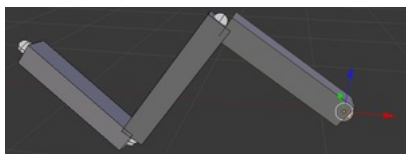
Luude visualiseerimiseks on neli põhilist viisi: oktaeeder, pulk, B-luu ja ümbrik:



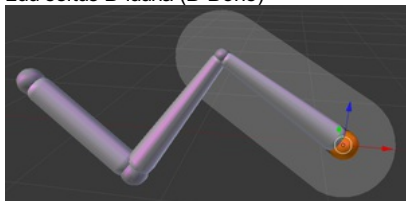
Oktaeedriline esitus (Octahedral).



Luu esitus pulgana (Stick)



Luu esitus B-luuna (B-Bone)



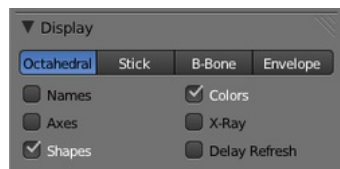
Luu esitus ümbrikuna (Envelope).

Esituse paneel (*Display*)

Mode: Objektirežiim Object, muutmisrežiim Edit ja poseerimisrežiim Pose

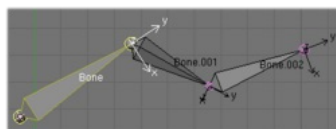
Panel: Esituse paneel Display objektiandmete kontekstis Object Data

Vaatame kõigepealt mõningaid üldisi visualiseerimise seadeid esituse paneelis Display objektiandmete kontekstis Object Data.



Esituse paneel Display.

Luu tüübid



Lihtne skelett oktaeedrilise esitusega (Octahedron), muutmisrežiim Edit.

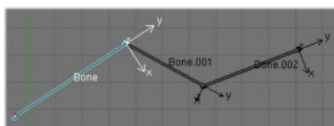
Pane tähele **40°** pööratud luud

Bone.001.

Oktaeedriline luu (*Octahedron*)

See on vaikimisi esitusrežiim, mis sobib hästi enamikeks muutmistoiminguteks. See näitab visuaalselt:

- Luu juurt ("järe" ots) ja tippu ("peenike" ots).
- Luu "suurust" (luu jämedus on proportsionaalne tema pikkusega).
- Luu pööret (kuivõrd tal on ruudukujuline ristlõige).

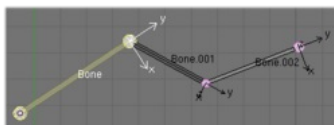


Sama skelett, kui esitusviisiks on pulk (Stick) poseerimisrežiimis Pose.

Pane tähele, et luu `Bone.001` pöördenuk ei ole nähtav (välja arvatud tema XZ-telgede asendi järgi).

Pulgakujuline luu (Stick)

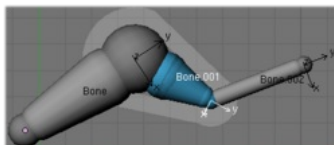
See on kõige lihtsam ja kõige vähem silmatorkav esitusviis. See näitab luid ühtlase (ja väikese) paksusega pulkadena ja seega ei anna teavet juure ega tipu asendite, luu suuruse ega tema pöörde kohta.



Sama skelett, kui esitusviisiks on B-luu (B-bone), muutmisrežiimis Edit.

B-luu esitus (B-Bone)

Selles esitusviisis on näha "sujuvate" mitmesegmendiliste luude kõverus; vaata täpsemalt [luude leheküljelt](#).



Luugruppide paneel Bone Groups.

Ümbriku esitus (Envelope)

See esitusviis näitab luu mõju objekti deformatsioonile. Loe selle kohta lähemalt [mõju leheküljelt](#).

Täiendavad esitused

Nimed (Names)

Kui see on sisse lülitatud, näidatakse 3D-vaates kõigi luude nimesid.

Värvid (Colors)

See omab tähendust ainult poseerimisrežiimis Pose ja tema kohta on saab täpsemalt lugeda [siit](#).

Teljed (Axes)

Kui see on sisse lülitatud, siis näidatakse luude (kohalikke) koordinaattelgi (ainult muutmisrežiimis Edit ja poseerimisrežiimis Pose).

Röntgen (X-Ray)

Kui see on sisse lülitatud, siis joonistatakse skeleti luud muidu läbipaistmatute objektide (võred, pinnad jms) peale, st nad on alati näha ja neid saab valida (see on sama valik, mis esituse paneelis Display objektandmete kontekstis Object Data). See on väga praktiline, kui 3D-vaate esitusrežiim ei ole traatvõre (Wireframe).

Kujud (Shapes)

Kui see on sisse lülitatud, siis objektirežiimis Object ja poseerimisrežiimis Pose asendatakse vaikimisi luu kuju valitud objekti kujuga (vaata lähemalt [altpoolt](#)).

Värskenduse viivitus (Delay Refresh)

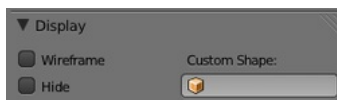
Kui see on sisse lülitatud, siis alluvaid poseerimisrežiimis ei deformeerita

Määratud kujuga luud

Mode: Objektirežiim Object ja poseerimisrežiim Pose

Panel: Esituse paneel Display luu kontekstis Bone

Blender võimaldab sul anda igale skeleti luule (objektirežiimis Object ja poseerimisrežiimis Pose) enda määratud kuju, kasutades mõnda olemasolevat objekti "mallina". Kõigepealt pead sa lülitama skeletipaneelil Armature sisse kujude nupu Shapes.



Esituse paneel Display.

Seaded

Traatvõte (Wireframe)

Kui see on sisse lülitatud, näidatakse luud sõltumata 3D-vaate esitusrežiimist traatvõrena. See on kasulik, et luude kujud ei varjaks vaadet.

Peida (Hide)

Luu on teistes režiimides peale muutmisrežiimi Edit nähtamatu.

Määratud kuju (Custom Shape)

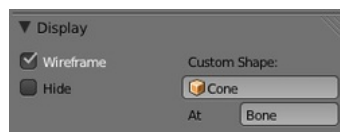
Objekt, mis määrab valitud luu kuju.

Määratud koht (Custom At)

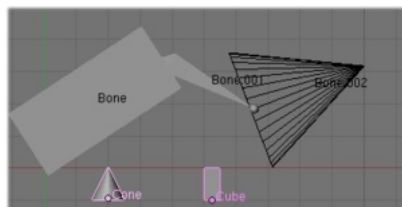
Luu, mis määrab esituses selle kuju luu teisenduse.

Et määrata luule eriline kuju:

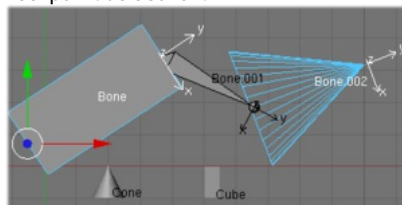
- Mine poseerimisrežiim Pose (Ctrl+ Tab).
- Vali soovitud luu (klõpsa temal RMB .
- Mine esituse paneelis Display määratud kuju väljale Custom Shape ja vali seal stseenis eelnevalt loodud 3D-objekt – selles näites kasutame me kuupi ja koonust. Lisaks võid sa väljal At määrata mõne teise luu.



Esituse paneel Display.



Skelett, kus kahele luule on määratud spetsiaalsed kujud objektirežiimis Object. Pane tähele objektide Cone ja Cube keskpunktide asukohti.



Sama skelett poseerimisrežiimis Pose

Pane tähele, et:

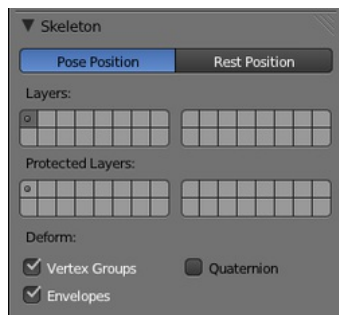
- Neid kujundeid ei renderdata kunagi – nagu kõik luud, on nad nähtavad ainult 3D-vaates.
- Kuigi väljal OB saab määrata suvalist tüüpi objekti (võre, kõvera, isegi metaobjekti), töötavad reaalselt ainult võred – kõik ülejäänud muudavad luu lihtsalt nähtamatuks, st ekraanile ei joonistata mitte midagi.
- Kujuna kasutatava objekti kese paikneb *luu juure* kohal (vaata juure ja tipu kohta täpsemalt [luude leheküljelt](#)).
- Vastava kuju objektiomadusi ignoreeritakse (st kui sa teed kuubist objektirežiimis Object mõõtkava teisendamise abil risttahuka, on luu kujuks ikka kuup).
- Luu "pikitelg" on kuju Y-telg ja objekti mõõtkava on alati teisendatud, nii et luu kogu pikkus juurest tipuni võtab täpselt ühe Blenderi ühiku.
- Kui sa tahad enda määratud luu kuju eemaldada, klõpsa lihtsalt parema hiirenupuga määratud kuju väljal Custom Shape ja vali hüpikmenüüst Reset to default value (taasta vaikekuju).

Kokkuvõttes – luu kujudena tuleb kasutada võresid, mille nullpunkt on luu juure kohal ja pikkus Y-telje suunas **1.0 BU**.

Skeletikihid

Mode: Objektirežiim Object, muutmisrežiim Edit ja poseerimisrežiim Pose

Panel: Skeletipaneel Skeleton, objektiandmete kontekst Object Data




Skeletipaneel Skeleton.


Igal skeletil on 32 “skeletikihti” (*Armature layers*), mis võimaldavad luid hallata, “grupeerides” need eri kihtidele. See töötab sarnaselt stseeni kihtidele (millel asuvad objektid). Sa saad “tõsta” luu teatud kihile, peita mõne kihi või näidata seda jne.

Luukihtide näitamine ja peitmine

Ainult aktiivsetel kihtidel asuvad luud on nähtavad ja muudetavad, kuid poseerimisel (st objektide liigutamisel või geomeetria deformeerimisel) töötavad kõik luud, olgu nad aktiivsetel või inaktiivsetel kihtidel. Kihi aktiveerimiseks või deaktiveerimiseks on mitu võimalust, sõltuvalt sellest, millises režiimis sa oled:

- Kõigis režiimides saab kasutada tillukeste nuppude rida skeleti paneeli Armature esituse sätete grupi Display Options ülaosas. Kui sa tahad mitut kihti korraga sisse või välja lülitada, hoi nuppudele klõpsamise ajal all klahvi ⇧ Shift.
- Muutmisrežiimis Edit ja poseerimisrežiimis Pose saad sa seda teha ka 3D-vaates, kas menüüst (Armature » Switch Armature Layers (skelett -> vaheta skeletikihte) või Pose » Switch Armature Layers) (poos -> vaheta skeletikihte) või kiirklahviga ⇧ ShiftM. Sellega avaneb väike dialoog, kus on needsamad kihtide nupud, nagu ülalpool mainitud (ka siin võid sa klõpsamise ajal hoida all klahvi ⇧ Shift LMB , kui tahad mitut kihti korraga valida).

Kaitstud kihid

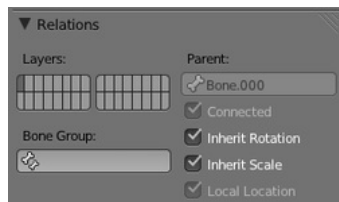
Sa saad mõne luude kihi kõigi oma skeleti [proksiobjektide](#) jaoks lukustada, st sellel kihid paiknevate luude parameetreid ei saa muuta. Selle tegemiseks klõpsa skeletipaneelil Skeleton vastaval nupul Ctrl LMB , lülitades lukustuse sisse.

Kaitstud kihtide proksiobjektil taastatakse proksiobjekti seaded faili uuesti laadimisel ja tegevuse tühistamisel.

Luukihid


Mode: Objektirežiim Object, muutmisrežiim Edit ja poseerimisrežiim Pose

Panel: Suhete paneel Relations luukontekstis Bone



Suhete paneel Relations.

Luude liigutamine kihtide vahel

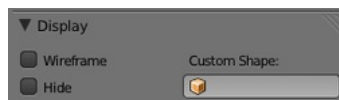
Nagu arvata on, pead sa luude kihilt kihile tõstmiseks olema muutmisrežiimis Edit või poseerimisrežiimis Pose – pane tähele, et nagu objektid, võivad ka luud olla korraga mitmel kihil, lihtsalt kasuta selleks ⇧ Shift LMB  klõpsamist, nagu ikka. Kõigepealt aga pead sa olema vajalikud luud valinud.

- Et määrata iga luu jaoks kihid, kuhu ta kuulub, kasuta nuppude aknas vastava luu “alampaneeli” “kihinuppe” (skeletiluude paneelis Armature Bones).
- 3D-vaates ava menüüga (Armature » Move Bone To Layer (skelett -> tõsta luu kihile) või Pose » Move Bone To Layer) (poos -> tõsta luu kihile) või kiirklahviga M kihinuppude dialoog. Pane tähele, et sel viisil *paigutad sa kõik valitud luud samale kihile*.

Luude peitmine

Mode: Muutmisrežiim Edit ja poseerimisrežiim Pose

Panel: Esituse paneel Display luukontekstis Bone



Esituse paneel Display.

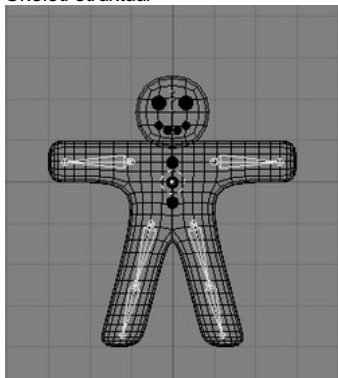
Luude näitamiseks või peitmiseks ei pea kasutama luude kihte. Nagu ka objektide, tippude või juhtpunktide puhul, võid sa kasutada kiirklahvi H:

- H peidab valitud luud.
- ⇧ ShiftH peidab kõik luud *peale valitute*.
- AltH näitab jälle kõiki peidetud luud.

Sa võid kasutada ka peitmise nuppu Hide luukonteksti Bone esituse paneelis Display.

Pane tähele, et peidetud luud on režiimist sõltuvad, st sa võid peita mõned luud muutmisrežiimis Edit, kuid nad on ikka nähtavad poseerimisrežiimis Pose ja vastupidi. Poseerimisrežiimis Pose peidetud luud on ka objektirežiimis Object nähtamatud. Ja lisaks peab muutmisrežiimis Edit luu peitmiseks olema valitud terve luu, mitte ainult tema juur või tipp.

Skeleti struktuur



[Piparkoogimehikese õppetüki](#) väga lihtne skelett.

Nagu öeldud, jälgendavad skeletid päris skelette. Nad koosnevad luudest, mis on (vaikimisi) jäigad elemendid. Sul on nende muutmiseks aga rohkem võimalusi kui reaalsete skelettide puhul. Lisaks "loomulikule" luude pööramisele saad sa neid ka nihutada ja isegi nende mõõtkava muuta! Ja sinu skeleti luud ei pea olema üksteisega ühendatud: kui soovid, võivad nad kõik olla omaette. Siiski on kõige loomulikumates ja praktilisemates skelettides osad luud üksteisega seotud, moodustades skeletis niinimetatud "luude ahelaid" (*Chains*) või mingit sorti "jäsemeid", nagu kirjeldatakse [allpool](#).

Luude ahelad (*Chains*)



Skelett kahe luude ahelaga.

Luud skeleti sees võivad olla üksteisest täiesti sõltumatud (st ühe muutmine ei mõjuta teisi), aga tihti ei ole see skeleti ülesehitus praktiline! Et teha midagi jala skeleti sarnast, eeldad sa, et kõik luud, mis tulevad "pärast" reieluud, liiguvad "koos" sellega, nagu oleksid nad selle küljes. Täpselt nii tehakse ka skelettides: "allutades" ühed luud teistele, saad sa teha "jäsemeid", mida Blenderis nimetatakse "luude ahelateks". Need ahelad saavad loomulikult ka hargneda, näiteks on kõik viis sõrme allutatud ühele "kämbaluule".

Luud paigutatakse ahelasse ühendades ülema luu tipu alluva luu juurega. Juur ja tipp võivad olla *seotud*, see tähendab, et nad asuvad alati täpselt samas punktis, või *vabad*, nagu tavalises hierarhilises suhtes.

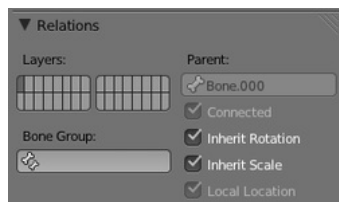
Üks ja sama luu võib olla mitme alluva ülem ja seega olla korraga osa mitmest ahelast.

Luud ahela alguses nimetatakse selle "juurmiseks luuks" (*Root Bone*) ja ahela viimast luud selle "tipuluuks" (*Tip Bone*) (ära neid termineid luude otspunktide nimedega segi aja).

Luude ahelad on eriti oluline teema [poseerimisel](#) (eriti standardses *otsekinemaatikaga* (*Forward Kinematics*) poseerimises, võrreldes "automaatse" *pöördkinemaatikaga* (*Inverse Kinematics*) poseerimisega). Sa lood ja muudad ahelaid muutmisrežiimis Edit, kuid kui luud ei ole seotud, ei mõjuta hierarhilised suhted selles režiimis luude teisendusi (st ülema teisendamine ei teisenda tema alluvaid).

Luude hierarhiliste suhete määramine

Seda kirjeldatakse pikemalt [muutmise peatükis](#), kuid heidame ka siin kiire pilgu sellesse olulisse teemasse.



Skeletiluude paneel Armature Bones - valitud on kaks luud ja nende allumissätted Child of on esile tõstetud.

Kõige lihtsam viis luude hieraria haldamiseks on suhete paneel Relations, mis asub luu kontekstis Bone:

- Kõigepealt, [vali](#) luud, mida sa tahad muuta (järjekord ei ole siin oluline).

Et üks luu teisele *allutada*, vali tema ülema nimi rippnimekirjast Parent. Et luud alluvussuhtest *vabastada*, vali lihtsalt samast ülema rippnimekirjast Parent tühi rida.

- Et luud ülema külge *siduda*, lülita ühendamise nupp Connected sisse.
- Et luud *lahi siduda*, lülita ühendamise nupp Connected välja.

Skeleti elementide valimine

Mode: Muutmisrežiim (Edit Mode)

Panel: Luu paneel Bone

Sa saad skeleti luid valida (ja muuta) nii muutmisrežiimis Edit kui ka poseerimisrežiimis Pose. Selles peatükis me näitame, kuidas valida luid muutmisrežiimis, kuid poseerimisrežiimis on see väga sarnane (üksikuid erinevusi kirjeldatakse detailiselt [poseerimise peatükis](#)).

Nii nagu võre servade valimiseks, on sul muutmisrežiimis Edit ka luude valimiseks kaks viisi:

- vali luud otse
- vali mõlemad selle otsad

täpselt nagu võre [tippude/servade valimisel](#).

Sellest on oluline aru saada, sest luu otspunktide valimisel võib programmi käitumine (vastavalt sellele, millise luu sa oled tegelikult valinud) olla mitte eriti ilmne.

Samuti pane tähele, et erinevalt võre esitusviisist ei mõjuta skeleti esitusviis valimist – luu tipu valik toimib alati samamoodi, sõltumata valitud luude esitusviisist.

Luu otspunktide valimine

Luu otspunktide valimiseks on sul kaks standardmeetodit.

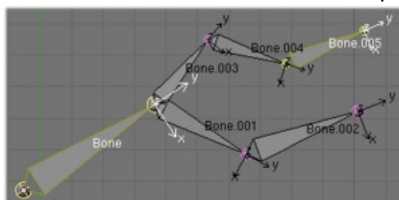
tegevus	kiirklahv	menüü	hiir
Luu otspunkti valimine			Vajuta sellel klahviga RMB
Lisa või eemalda valikust			⇧ Shift RMB
Vali või tühistä kõigi otspunktide valik	A	Select » Select/Deselect All (valik -> vali/tühista kõik)	
Pööra valik vastupidiseks	Ctrl	Select » Inverse (valik -> vastupidine)	
Piirdkastiga valiku sisselülitamine	B	Select » Border Select (valik -> piirdkast)	klõpsa LMB ja lohista kast ümber nende otspunktide, mida tahad valida klõpsa LMB ja lohista, et valikust eemaldada kinnituseks lase LMB lahti tühistamiseks vajuta Esc või klõpsa RMB
Piirdkastiga valimine			RMB
Piirdkastiga valiku välja lülitamine	B või Esc		klõpsa Ctrl LMB ja lohista lasso ümber nende otspunktide, mida tahad valida klõpsa Ctrl⇧ Shift LMB ja lohista valikust eemaldamiseks kinnituseks lase LMB lahti tühistamiseks vajuta Esc või klõpsa RMB
Lassoga valik			

Valiku vastupidiseks pööramine

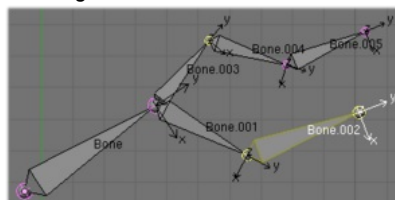
Nagu eespool öeldud, pead sa meeles pidama, et need valikutööriistad on ainult luude otspunktide valimiseks, vastasel korral võid sa tulemusest segadusse sattuda.

Näiteks valiku ümber pööramine Inverse (Ctrl) pöörab ümber luude otspunktide valiku, mitte luude endi valiku (vaata pilti *Valiku vastupidiseks tegemine*).

Valiku vastupidiseks tegemine



Valitud on kaks luud.



Valiku ümber pööramise (Ctrl) tulemus: vahetatud on luude otste, mitte luude endi valik.

Seotud luude otste valimine

Veel üks näide: kui sa valid ülemaga seotud luu juure, siis sa valid automaatselt ka ülema tipu (ja vastupidi).


Pea meeles: luude otste valimisel on seotud ülema tipp "seesama asi", mis temale allutatud luude juured.

Luude valimine

Klõpsates RMB luu "kehal", valid sa terve luu (ja seega ka automaatselt tema juure ja tipu).

Igale valitud luule vastab skeetiluu paneelis Armature Bones alampaneel (muutmiskontekstis Editing, F9). Nendel alampaneelidel,

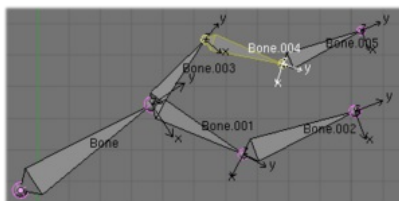
nagu me hiljem näeme, on teatud luu omaduste seaded (mis puudutavad näiteks luude hierarhilisi suhteid, luude mõju deformeeritavale geomeetria jne.)

Klõpsates  Shift RMB, saad sa luid valikule lisada või valikust eemaldada.

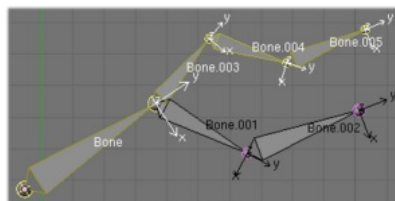
Lisaks on veel mõned **keerukamad valikuvahendid**, mis tuginevad luude hierarhilistele suhetele.

Sa saad valida korraga kõik luud ahelas, kuhu kuulub aktiivne (viimati valitud) luu, kasutades *ühendatud valiku* tööriista (L, *linked selection*).

Ühendatud luude valimine



Üks valitud luu.



Kiirklahviga L on valitud terve selle ahel.

Sa saad aktiivse luu valikust eemaldada ja samaaegselt valida tema ülema või ühe tema alluvatest, kasutades menüüd Select » Select Parent (valik -> vali ülem) (I) või Select » Select Child (valik -> vali alluv) (J). Kui sa eelistad jätta aktiivse luu valikusse, kasuta Select » Extend Select Parent (valik -> vali lisaks ülem) (Ctrl I) või Select » Extend Select Child (valik -> vali lisaks alluv) (Ctrl J).

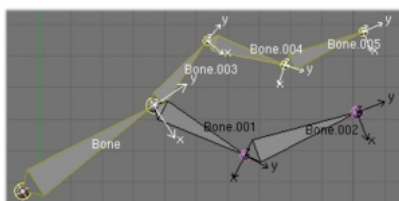
Seotud luude valikust eemaldamine

Seotud luudega seoses on üks nüanss.

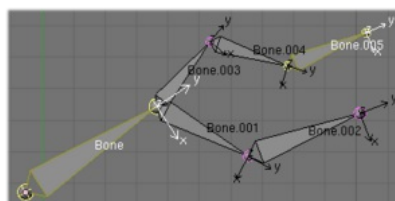
Kui sa oled valinud mitu omavahel seotud luud, siis ühe luu valikust eemaldamisel *eemaldad sa tegelikult valikust ainult selle tipu, aga mitte juurt, kui viimane on samal ajal teise valitud luu tipuks*.

Vaata pilti *Valitud ahelasse kuuluva luu eemaldamine valikust*.

Valitud ahelasse kuuluva luu eemaldamine valikust



Valitud ahel.



Pärast  Shift RMB klõpsamist Bone.003

Pärast  Shift RMB klõpsamist Bone.003:

- Bone.003 tipp on valikust eemaldatud ja koos sellega ka Bone.004 juur
- Bone.003 juur on endiselt valitud, sest Bone on Bone.003 ülem ja jäi valituks.

Skeleti muutmine

Mode: Muutmisrežiim (Edit Mode)

Hotkey: ⇆ Tab

Nagu igat teistki objekti, saab skeletti muuta muutmisrežiimis Edit (⇆ Tab).

Skeleti muutmine tähendab kahte tegevust:

- [Luude muutmine](#) – st nende lisamine/kustutamine/eendamine/jagamine/liitmine.
- [Luude omaduste muutmine](#) – sealhulgas luude põhiliste omaduste, nagu teisendus (st asukoht, mõõtkava jne), luudevaheliste suhete (allutamine ja sidumine) ja ka luude nimede, mõju, poseerimisrežiimis Pose käitumise jne muutmine.

Need on tavalised muutmisvõimalused, mis samanevad näiteks vastavate toimingutega [võre](#) muutmisel. Blenderis on ka palju keerukam “skeleti visandamise” tööriist, mille nimi on [Etch-a-Ton](#). Sama tööriista abil saab ka skelette [matkida](#), st kasutada ühte skeletti teise mallina.



Üks oluline asi, millest skeleti muutmisel tuleb aru saada, on see, et sa **muudad oma skeleti puhkepositsiooni (Rest Pose)**, st tema “vaikeolekut”. *Puhkepositsioonis* on skeleti kõik luud oma kohaliku koordinaatsüsteemi suhtes ilma pöördeta ja skaalaga **1.0**.

Erinevad [poosid](#), mida sa hiljem võid luua, põhinevad sellel puhkepositsioonil – seega, kui sa muudad skeletti muutmisrežiimis Edit, muutuvad ka kõik ülejäänud poosid. Seega peaksid sa üldiselt olema kindel, et skelett on valmis, enne kui hakkad seda [nahaga katma](#) ja [poseerima](#)!



Pane tähele, et mõned tööriistad töötavad luude otspunktidega ja mõned luude endiga. Ole hoolikas, et mitte segadusse sattuda.

Luude muutmine


Mode: Muutmisrežiim (Edit Mode)

Hotkey: ⇧ Tab

Selles peatükis vaatame, kuidas Blenderis luid [lisada](#), [kustutada](#) ja [osadeks jagada](#). Samuti seda, kuidas [blokeerida luude teisendamist](#) muutmisrežiimis Edit ja kuidas skeleti loomisel kasutada [automaatset peegeldust](#).

Luude lisamine

Skeletile luude lisamiseks on samad võimalused nagu võre muutmisel:

- Lisamismenüü Add,
- Eendamine (*Extrusion*),
- Ctrl LMB  klõpsamine,
- Lisamine liigeste vahele,
- Duplitseerimine.

Lisamismenüü (*Add*)

Mode: Muutmisrežiim (Edit Mode)

Hotkey: ⇧ ShiftA

Skeletile uue luu lisamiseks vajuta 3D-vaates ⇧ ShiftA » Bone (luu).

Nii loodud luu on:

- ühe Blenderi ühiku pikkune,
- sihitud vaate Y-telje positiivses suunas,
- lähtepunktiga 3D-kursori asukohal
- ei ole seotud ühegi teise skeleti luuga.

Eendamine (*Extrusion*)

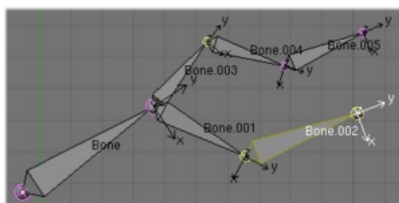
Mode: Muutmisrežiim (Edit Mode)

Hotkey: E, ⇧ ShiftE

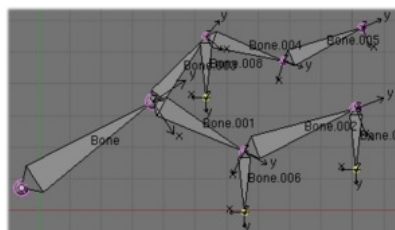
Menu: Armature » Extrude (skelett > enda)

Kui vajutad kiirklahvi E, loob Blender igast (otseselt või kaudselt) valitud luutipust lähtuva uue luu. Uus luu on allutatud "selle" tipu omanikule ja on sellega ühendatud. Nagu ikka, jäävad pärast eendamist valituteks ainult uute luude tipud ja aktiveeritakse nihutamisrežiim, et saaksid luud kohe soovikohaselt paigutada. Vaata *Eendamise näidet*.

Eendamise näide



Skelett kolme valitud luutipuga.



Kolm eendatud luud.

Saad eendamisel kasutada ka mõõtkava teisendamist ja pööramist, nagu kirjeldatud võre muutmise [peatükis](#), vajutades vastavalt ER või ES – samuti võid "[lukustada](#)" eendamise ühele globaalsele või lokaalsele teljele.

Luudel on unikaalne "peegeldusega eendamise" tööriist, mille kasutamiseks vajuta ⇧ ShiftE. Vaikimisi töötab ta täpselt nagu tavaline eendamine. Kui sa aga oled X-telje peegelduse muutmisvaliku ("X-Axis mirror") sisse lülitanud (vaata täpsemalt [altpoolt](#)), tekib igast eendatud tipust *kaks uut luud*, millel on sama nimi, välja arvatud erinev "_L/_R" kood lõpus (vastavalt siis vasaku ja parema luu jaoks - sellest loe täpsemalt [järgmiselt leheküljelt](#)). Luu koodiga "_L" käitub täpselt samal viisil nagu vaikimisi seadetega eendatud luu – teda saab nihutada, pöörata ja selle mõõtkava teisendada. Luu koodiga "_R" on tema peegelpildis paariline (piki skeleti kohalikku X-telge) – vaata pilti *Peegeldusega eendamise näide*.



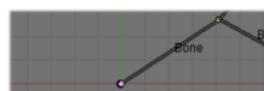
Pane tähele, et täpselt nagu võre muutmisel, kui vajutad kohe pärast E vajutamist Esc, jäävad eendatud luud alles, kuid nende pikkus on 0 – see võib sulle hiljem probleeme tekitada. Kui märkad seda kohe, saad eendamise tühistada, vajutades CtrlZ.

Muuseas: erinevalt võre muutmisest ei saa sa luudest vabaneda lihtsalt X vajutades, sest eendamisel valitakse uute eendatud luude tipud, ja nagu allpool selgitatakse, ei tööta kustutuskäsk, kui valitud on ainult luutipud. Selliste eendatud luude kustutamiseks ennistamist kasutamata pead nende tippe veidi liigutama, seejärel valima terved luud ja need [kustutama](#).

Hiirenupud

Mode: Muutmisrežiim (Edit Mode)

Peegeldusega eendamise näide



Üks valitud luutipp




Kaks peegeldusega eendatud luud

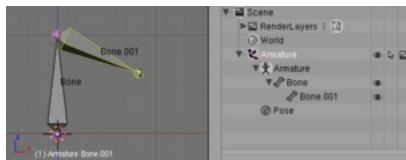
Hotkey: Ctrl LMB 

Kui vähemalt üks luu on valitud, lisab kombinatsioon Ctrl LMB  skeletile uue luu.

Märkused uue luutipu kohta:

- Pärast Ctrl LMB  klõpsu saab sellest skeleti aktiivne element.
- Luu ilmub sinna, kuhu klõpsasid, kuid...
- ... (nagu ka võre muutmisel) ta on paigutatud tasapinnale, mis on paralleelne vaate tasandiga, sügavuti 3D-kursorit läbides.

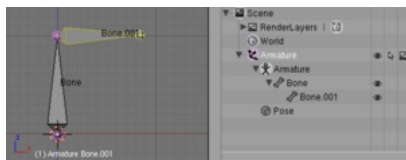
Luu juure asend ja paiknemine teiste luude hierarhias sõltub aktiivsest elemendist:



Ctrl-klõps, kui aktiivseks elemendiks on **luu**

Kui aktiivne element on **luu**, siis:

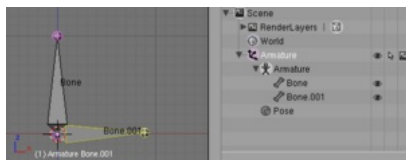
- uue luu juur paigutatakse aktiivse luu tipuga kohakuti
- uus luu on hierarhias aktiivse luu alluv ning temaga seotud (vaata ülevaateakent näites *Ctrl-klõps, kui aktiivseks elemendiks on luu*).



Ctrl-klõps, kui aktiivseks elemendiks on **tipp**

Kui aktiivne element on **tipp**, siis:

- uue luu juur paigutatakse aktiivse tipuga kohakuti
- uus luu on hierarhias aktiivse tipu omaniku alluv ning temaga seotud (vaata ülevaateakent näites *Ctrl-klõps, kui aktiivseks elemendiks on tipp*).

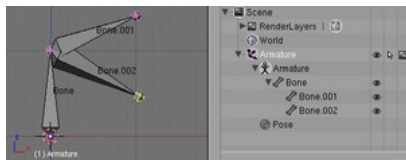


Ctrl-klõps, kui aktiivseks elemendiks on **lahtine juur**

Kui aktiivne element on **lahtine (sidumata) juur**, siis:

- uue luu juur paigutatakse aktiivse juurega kohakuti
- uut luud **EI** allutata hierarhias aktiivse juure omanikule (vaata ülevaateakent näites *Ctrl-klõps, kui aktiivseks elemendiks on lahtine juur*).

Seega **ei** ole see uus luu ühendatud ühegi teise luuga.



Ctrl-klõps, kui aktiivseks elemendiks on **seotud juur**

Kui aktiivne element on **seotud juur**, siis:

- uue luu juur paigutatakse aktiivse juure kohale
- uus luu **allutatakse** hierarhias aktiivse juure omaniku ülemale ja seostatakse temaga (vaata ülevaateakent näites *Ctrl-klõps, kui aktiivseks elemendiks on seotud juur*).

See peaks olema ilmne, sest kui aktiivne element on seotud luu juur, on see element samal ajal ka tema ülemluu tipp ning seega käitub programm samaselt teisele näitele.

Kuivõrd uue luu tipust saab aktiivne element, saad teha järjest terve seeria Ctrl-klõpse, lisades iga kord uue luu sama luude ahela lõppu.

Liigeste vahe täitmine (*Fill between joints*)

Mode: Muutmisrežiim (Edit Mode)

Hotkey: F

Menu: Armature » Fill Between Joints (skelett -> täida liigeste vahe)

Seda tööriista kasutatakse põhiliselt kahe valitud tipu vahele uue luu loomiseks (F), analoogiliselt "servade/külgede loomisele" võre muutmisel.

Kui valitud on üks juur ja üks tipp, siis uue luu:

- juur paigutatakse valitud tipu kohale
- tipp paigutatakse valitud juure kohale
- luu allutatakse hierarias tipu omanikule ja seotakse sellega

Tipu ja juure vahe täitmine



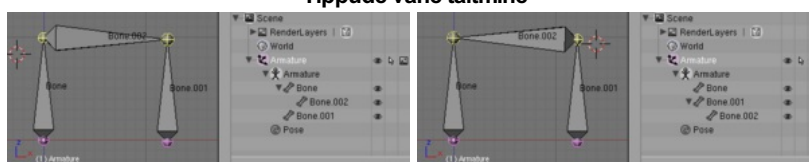
Aktiivne tipp vasakul

Aktiivne tipp paremal

Kui valitud on kaks tippu, siis uue luu:

- juur paigutatakse selle valitud tipu kohale, mis on 3D-kursorile lähemal
- tipp paigutatakse teise valitud tipu kohale
- luu allutatakse hierarias ja seotakse selle luuga, mille tipu kohale paigutati uue luu juur

Tippude vahe täitmine



3D-kursor vasakul

3D-kursor paremal

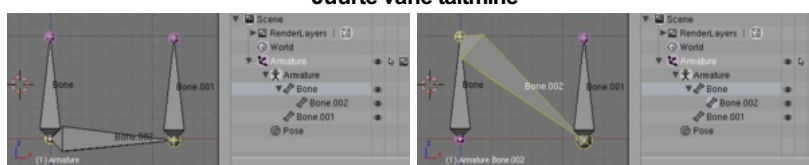
Kui valitud on kaks juurt, tekib väike probleem, sest Blenderi sisemine teadeteahel ei uuenda kasutajaliidest reaalselt.

Kui sarnaselt eelmisele näitele vajutada F, siis näed uut luud, mille:

- juur paigutatakse selle valitud juure kohale, mis on 3D-kursorile lähemal
- tipp paigutatakse teise valitud juure kohale
- luu allutatakse hierarias ja seotakse selle luuga, mille juure kohale paigutati uue luu juur

Kui üritad nüüd uut luud liigutada, siis 3D-vaadet värskendatakse ja sa näed uue luu juurt hüppamas tema ülemä tipu kohale.

Juurte vahe täitmine

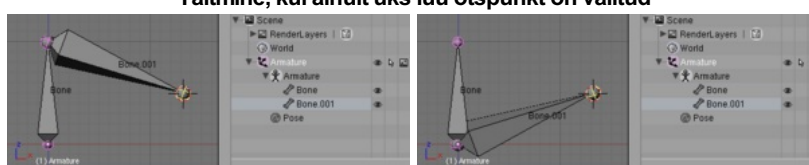


Enne vaate värskendust (3D-kursor vasakul)

Pärast vaate värskendust, õige esitus

Kui vajutada F siis, kui valitud on ainult üks luu otspunkt, luuakse uus luu valitud otspunkti ja 3D-kursori vahele ja seda ei allutata ühelegi skeleti luule.

Täitmine, kui ainult üks luu otspunkt on valitud



Täitmine, kui ainult üks tipp on valitud

Täitmine, kui ainult üks juur on valitud

Tulemuseks on viga, kui üritad:

- täita sama luu otspunktide vahet või
- täita rohkem kui kahe valitud tipu vahet.

Duplitseerimine (Duplication)

Mode: Muutmisrežiim (Edit Mode)

Hotkey: ⇧ ShiftD

Menu: Armature » Duplicate (skelett -> duplitseeri)



See tööriist mõjutab vaid valitud luid - valitud tippe ignoreeritakse.

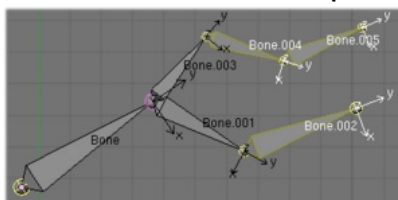
Nagu võre muutmiselgi, vajutades ⇧ ShiftD:

- tehakse valitud luudest koopiad,
- koopiad valitakse ja aktiveeritakse asukoha määramiseks nihutamisrežiim.

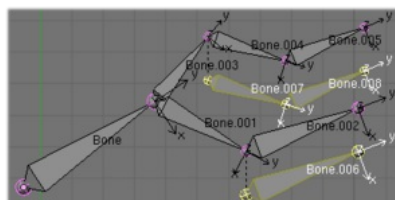
Kui oled valinud ahela osa, siis selle duplitseerimisel tehakse koopia kogu ahelast, nii et duplikaadid on omavahel ühendatud originaaliga analoogselt.

Hierarhias teisele luule allutatud luu duplikaat on samuti samale luule allutatud – isegi siis, kui tema ülem ei olnud duplitseerimiseks valitud. Pane aga tähele, et kui luu on allutatud **ja seotud** valimata luuga, siis tema duplikaat on küll sellele allutatud, **aga mitte seotud** (vaata pilti *Duplitseerimise näide*).

Duplitseerimise näide



Skelett kolme valitud luu ja ühe valitud juurega



Kolm duplitseeritud luud. Pane tähele, et valitud ahel säilib duplikaadis ja Bone.006 (ja vastavalt ka Bone.007) on allutatud, aga mitte seotud luuga Bone.001 (ja vastavalt ka Bone.003), nagu märgivad mustad punktiirjooned.

Luude kustutamine

Luude eemaldamiseks skeletist on kaks viisi: tavaline kustutamine ja mitme luu omavaheline liitmine.

Tavaline kustutamine

Mode: Muutmisrežiim (Edit Mode)

Hotkey: X

Menu: Armature » Delete (skelett -> kustuta)



See tööriist mõjutab vaid valitud luid - valitud tippe ignoreeritakse.

Luu kustutamiseks võid sa:

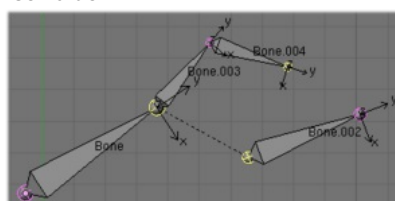
- vajutada standardset kiirklahvi X ja kustutamine kinnitada või
- valida menüüst Armature » Delete (skelett -> kustuta) ja see valik kinnitada.

Kui kustutad luu ahelast, siis temale alluvad luud allutatakse automaatselt kustutatud luu ülemale, kuid skeleti deformatsioonide vältimiseks **ei seota** luid omavahel.

Kustutamise näide



Skelett kahe valitud luuga hetk enne kustutamist



Kaks luud on kustutatud. NB! Luu Bone.002, mis oli enne allutatud kustutatud luule Bone.001, on nüüd allutatud (aga mitte seotud) luuga Bone.

Ühendamine (Merge)

Mode: Muutmisrežiim (Edit Mode)

Menu: Armature » Merge (skelett->ühenda)

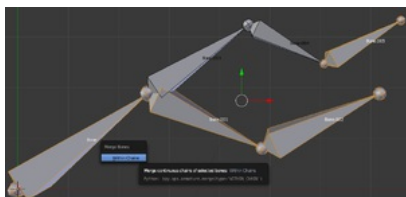
Võid ühendada korraga mitu luud tingimusel, et *nad moodustavad ahela*. Iga luudest moodustuv valitud ahel asendatakse ühe luuga, mille juureks on ahela esimese luu juur ja tipuks viimase luu tipp.

Kinnita ühendamine, valides kinnitusaknas Merge Selected Bones (ühenda valitud luud) sissekannet Within Chains (ahelate sees).

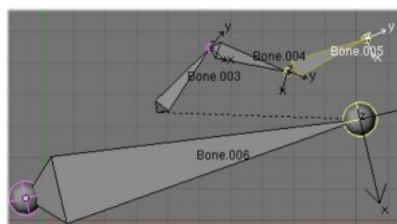
Kui mõni teine (valimata) ahel algab ühendatava ahela keskelt, allutatakse see ühendamise tulemusena moodustatud luule. Kui ta oli ülemaga seotud, seotakse ta uue luuga.

Siin on üks kummaline nüanss (vaata pilti *Ühendamise näide*): ehkki seotud (ühendamata ahela esimesel luul ei ole juure kera), ei ole luud omavahel nähtavalt ühendatud. Viimane toimub alles siis, kui sa muudad mõnda luud, ja tulemus sõltub sellest, millises ahelas muudetak luu asub (seda selgitavad illustratsioonid paremini).

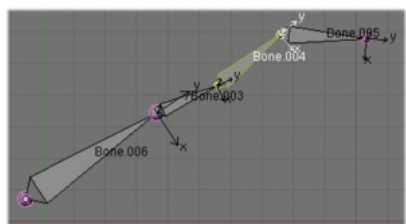
Ühendamise näide



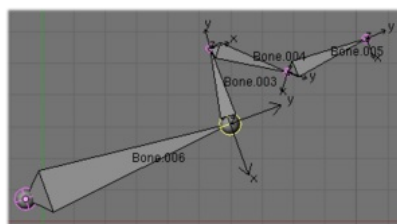
Skelett, milles on valitud luude ahel ja üks üksik luu hetk enne ühendamist



Luud Bone.001 ja Bone.002 on ühendatud luuks Bone.006, samas kui luu Bone.005 jäi muutumatuks. NB! Luu Bone.003, mis on seotud luuga Bone.006, ei ole veel "tõeliselt" ühendatud.



Luud Bone.004 pöörati ja sellega liikus luu Bone.006 luu Bone.003 juure asukohta.



Luu Bone.006 tippu liigutati ja sellega liikus Bone.003 luu Bone.006 tippu asukohta...

Luude tükeldamine (Subdividing)

Mode: Muutmisrežiim (Edit Mode)

Hotkey: W1, W2

Menu: Armature » Subdivide (skelett -> tükelda), Armature » Subdivide Multi (skelett -> tükelda mitmeks)

Ühest luust mitme tegemiseks võid selle tükeldada. See tööriist tükeldab kõik valitud luud olemasolevaid alluvussuhteid säilitades. Tükeldamise käigus loodud uued luud moodustavad alati omavahel seotud luude ahela.

Selleks, et jagada iga valitud luu kaheks:

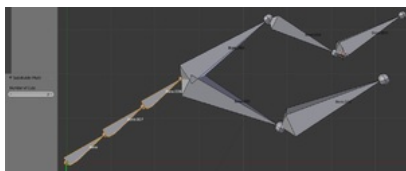
- vajuta W » Subdivide (tükelda) - sama mis W1 või
- vali päise menüüst Armature » Subdivide (skelett -> tükelda)

Selleks, et teha igast valitud luust suvaline arv uusi:

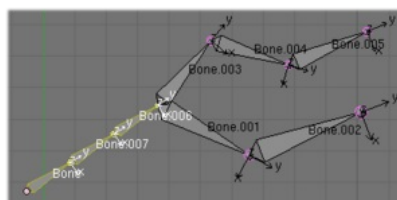
- vajuta W » Subdivide Multi (tükelda mitmeks) - sama mis W2 või
- vali päise menüüst Armature » Subdivide Multi (skelett -> tükelda mitmeks)

Seejärel määra ilmuvas dialoogis soovitud lõigete arv. Analoogselt võre muutmisele on lõigete arvu n puhul tulemuseks igast valitud luust $n+1$ uut luud.

Tükeldamise näide



Skelett ühe valitud luuga hetk enne tükeldamist



Valitud luud on "lõigatud" kaks korda.

Tekkis kolm uut luud.

Luude lukustamine

Luu muutmisrežiimis Edit teisendamist saad vältida mitmel viisil:

- Aktiivse luu saab lukustada, vajutades lukustamise nuppu Lock 3D-vaate teisenduse omaduste paneelis Transform Properties (N);
- Kõiki luid saab lukustada lukustamise nupu Lock abil, mis paikneb skeletiluude paneelis Armature Bones asuvatel alampaneelidel;
- Vajuta \varnothing ShiftW » Toggle Settings » Locked (vaheta seadeid -> lukustamine)
- Vali menüüst Armature » Bone Settings » Toggle a Setting (skelett -> luu seaded -> vaheta seadet).

Kui lukustatud luu juur on seotud lukustamata luu tipuga, siis seda ei lukustata - st sa saad juurt vabalt liigutada. See tähendab, et kui lukustad ühe luu seostatud luude ahelas, lukustad sa tegelikult ainult selle tipu. Sidumata luude puhul mõjub lukustamine nii tipule kui ka juurele.

X-pegeldusega (X-Axis Mirror) muutmine

Üks väga kasulik tööriist on X-pegelduse valik X-Axis Mirror (tööriistade paneel Tools > skeleti seaded Armature Options, kui muutmisrežiimis on skelett valitud), mis töötab veidi sarnaselt vastavale [võre tööriistale](#). Kui sul on luude paarid, mille nimed on ühesugused, välja arvatud "külge määrav lõppliide" (näiteks `.R.L` või `_right/_left...`) ja see valik on aktiveeritud, siis iga kord, kui mõnd luud teisendad (nihutad/pöörad/muudad mõõtkava...), siis tema "teise poole" paarilist teisendatakse samal viisil, kuid *skeleti kohaliku X-telje suhtes sümmeetriliselt*. Kuna enamikel skelettidel on vähemalt üks sümmeetriatelg (loomad, inimesed, ...) võimaldab see pool redigeerimise aega kokku hoida! Täpsemalt loe luude nimetamisest [järgmiselt leheküljelt](#).

Luude eraldamine uude skeletti

Nagu võrede puhul, saad ka skeleti valitud luud sõltumatuks skelettobjektiks eraldada (Armature » Separate (skelett -> eralda), CtrlAltP) – ja loomulikult saad sa objektirežiimis Object ühendada kõik valitud skeletid üheks (Object » Join Objects (objekt -> ühenda objektid), CtrlJ).

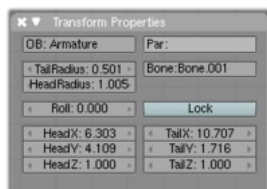
Luude omaduste muutmine

Selles peatükis me vaatame, kuidas muuta või määrata enamikke luude omadusi – sa peaks olema juba lugenud [eelmist peatükki](#)! Me näeme, kuidas [hallata luude hierahiat](#), [luid ümber nimetada](#) jne.

Luude teisendamine

Me ei keskendu siin erinevatele luude teisendustele ega täiendavatele võimalustele nagu telgede lukustamine, pöörlemiskese ja nii edasi, sest need on kehtivad enamike objektide muutmisel ja neid on juba kirjeldatud [selles peatükis](#) (pane ainult tähele, et mõned funktsioonid, nagu näiteks nakkumine, ei näi töötavat, ehkki nad on kasutajaliideses olemas). Sama kehtib ka peegeldamise kohta, mis on praktiliselt identne sama funktsiooniga [võre muutmisel](#). Lihtsalt võta teadmiseks, et luude otspunktid käituvad enam-vähem nagu võre tipud ja luud ise nagu võre servad.

Nagu sa tead, on luude vahel kahte tüüpi suhted: ühe luu saab allutada teisele ja allutatud luu saab ülemaga siduda. Muutmisrežiimis Edit käituvad alluvad luud täpselt nii, nagu nad ei oleks hierarhilises suhtes: sa võid nende ülemat nihutada, pöörata ja selle mõõtkava teisendada ilma alluvate asendit muutmata. Seevastu seotud luude puhul on alluva juur alati samas punktis, kus ülema tipp, nii et luud teisendades mõjutad sa temaga seotud ülemat, tema alluvaid ja kaaslasi.



Skeleti teisendussätete paneel Transform Properties muutmisrežiimis Edit

Lisaks saad sa teisendussätete paneelis Transform Properties (N) muuta mõlema luu otspunkti asukohta ja raadiust ning ka luu [pööret ümber oma telje](#).

Raadius ja mõõtkava ümbriku esitusrežiimis

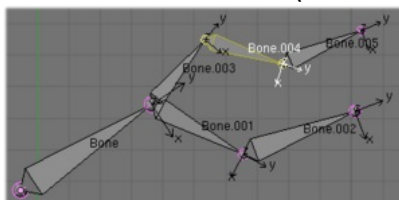
Mode: Muutmisrežiim Edit, ümbriku esitusviis Envelope

Hotkey: S

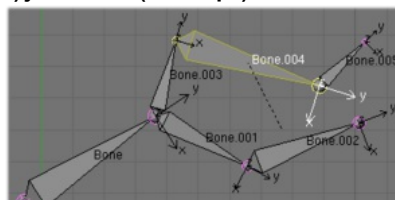
Menu: Armature » Transform » Scale (skelett -> teisenda -> skaleeri)

Ehkki mõõtkava teisendamine oktaeedri (Octahedron), pulga (Stick) ja B-luu (B-Bone) esitusviisides käitub oodatavalt, on tema mõju ümbriku esitusviisides (Envelope) erinev: luu enda asemel muudab ta valitud otspunktide raadiuste mõõtkava; vaata lähemalt [nahaga katmise peatükist](#). Kuivõrd sa muudad ainult üht suurust (raadiust), telgede lukustamine siin ei toimi. Ja nagu ikka, seotud luude mõõtkava teisendamisel muudad sa korraga ülema tipu ja alluva juure raadiusi.

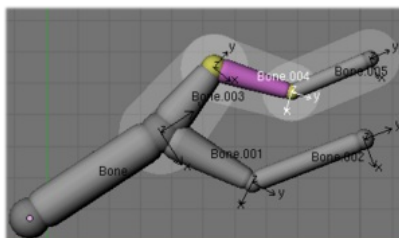
Luu skaleerimine oktaeedri (Octahedron) ja ümbriku (Envelope) esitusviisides.



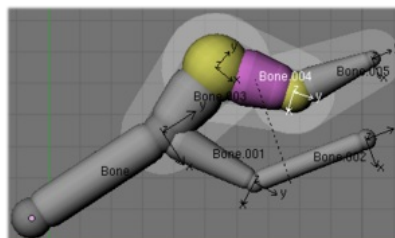
Üks valitud luu...



...mille mõõtkava on teisendatud oktaeedrilises esituses.



Üks valitud luu...



...mille mõõtkava on teisendatud ümbriku esitusviisides – luu pikkus jääb samaks, aga tipu raadius kasvab.

Pane tipu raadiuse puhul tähele, et kui sa luu suurust muudad (kas otseselt mõõtkava teisendades või luu otspunkte nihutades), muudab Blender tipu raadiust automaatselt proportsionaalselt muutuse suurusle. Nii et neid omadusi tasub häälestada alles siis, kui kõik luud on lõplikult paigas!

BMõõtkava (ScaleB) ja ümbrik

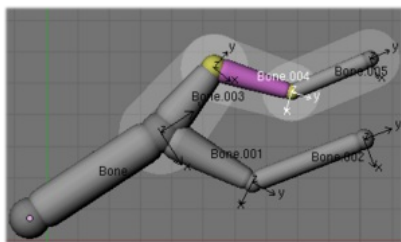
Mode: Muutmisrežiim (Edit Mode)

Hotkey: AltS

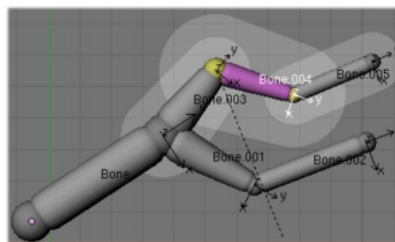
Kiirklahv AltS aktiveerib ühe ainult skeletile omase lisatööriista, mille täpne funktsioon sõltub esitusviisist.

Ümbriku esitusviis võimaldab see muuta valitud luude mõjuulatust (nende kauguse sätet Dist, vaata [nahaga katmise peatükki](#)) – nagu ka “tavaline” mõõtkava teisendamine selles esitusviisis, on see ühe parameetri teisendus ja ei toeta telgede lukustamist.

Ümbriku mõõtkava teisendamise näide



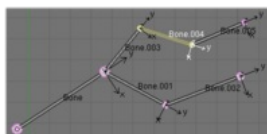
Üks ümbriku esitusviisis valitud luu



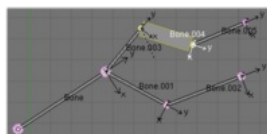
Tema ümbriku suurust on kiirklahviga AltS muudetud

Teistes esitusviisides võimaldab ta muuta “luu suurust”. Sellel näib olevat nähtav tulemus ainult B-luu esitusviisis, kuid ta on olemas ka oktaeedri ja pulga puhul. Sellel tööriistal on veel üks eriline omadus: kui kõigi teiste tööriistade jaoks on “kohalikud koordinaatteljed” objekti omad, siis antud juhul on nendeks luu omad (see tähendab, et kui sa kaks korda vastavat klahvi vajutades lukustad teisendustelje, kasutatakse luu kohalikku koordinaattelge, mitte skeleti oma). Muide, ära ürita lukustada selle tööriista kasutamisel koordinaattelgi, kui valitud on rohkem kui üks luu – see väga tõhus viis Blenderi kokkujooksutamiseks!

“Luu suuruse” mõõtkava muutmise näide



B-luu esitusviisis on valitud üks “vaikimisi suurusega” luu



Selle suurust on kiirklahviga AltS muudetud



Sama skelett objektirežiimis ja B-luu esitusviisis – Bone.004 mõõtkava on suurendatud

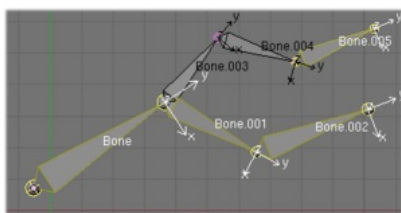
Luu suund

Mode: Muutmisrežiim Edit

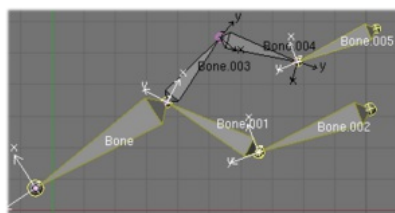
Hotkey: W3

Menu: Specials » Switch Direction (lisad -> vaheta suund)

See tööriist ei ole kättesaadav mitte skeletimenüüst Armature, vaid ainult lisade kontekstimenüüst Specials (W). See võimaldab valitud luude suunda vahetada nii, et juurest saab tipp ja vastupidi. *Luu suuna muutmine üldjuhul lõhub ahelad, mille osaks see luu on.* Siiski, kui sa muudad terve (või osalise) ahela suunda, jäävad muudetud luud ikkagi üksteisega seotuks, kuid “vastupidises järjekorras”. Vaata *Suunamuutuse näidet*.



Skelett valitud üksiku luu ja kolmeliulise ahelaga just enne suuna muutmist



Valitud luude suund on muudetud Bone.005 ei ole enam ei allutatud ega seotud ühegi luuga. Ümberpööratud luude ahel on alles, kuid vastupidises järjekorras (Bone.002 on nüüd ahela juur ja Bone tipp). Bone.003 on nüüd vaba luu.

Suunamuutuse näide.

Luu pööramine



Mode: Muutmisrežiim Edit

Hotkey: CtrlR, CtrlN

Menu: Armature » Bone Roll » ... (skelett -> luu pööre)

Muutmisrežiimis on valikud, mille abil saab muuta luu pööret ümber pikitelje (st ümber luu Y-telje). Iga kord, kui sa uue luu lisad, on ta paigutatud nii, et Z-telg oleks võimalikult risti hetke vaatesuunaga. Ja iga kord, kui sa luud teisendad, üritab Blender leida talle parimat pööret.

Selle tulemuseks võib aga olla ebakorrapärane skelett, milles luud on suvalistes suundades pööratud... mis ei ole kena! Selle probleemi lahendamiseks on kolm võimalust:

- Armature » Bone Roll » Set Roll (skelett -> luu pööre -> määrata pööre) (CtrlR) käivitab ümber pikitelje pööramise, mis käitub nagu iga teine objekti pööramine (st liiguta hiirt ja klõpsa LMB  kinnitamiseks või sisesta number ja vajuta enterit; katkestamiseks klõpsa RMB  või vajuta Esc).
- Armature » Bone Roll » Clear Roll (Z-Axis Up) (skelett -> luu pööre -> algväärtusta) (või CtrlN1 » Recalculate Bone Roll Angles » Clear Roll (Z-Axis Up) (Arvuta luu pööre ümber -> algväärtusta)) algväärtustab luu pöörde nii, et luu kohalik Z-telg langeks võimalikult palju kokku globaalse Z-teljega.
- Armature » Bone Roll » Roll to Cursor (skelett -> luu pööre -> pööra kursorini) (või CtrlN2 » Recalculate Bone Roll Angles » Align Z-Axis to 3D-Cursor (Arvuta luu pööre ümber -> joonda Z-telg 3D-kursoriga)) keerab luu sellisesse suunda, et tema kohalik Z-telg oleks võimalikult 3D-kursori suunas.

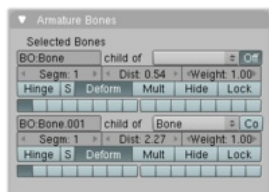
Omadused

Mode: Muutmisrežiim Edit

Panel: Skeletiluude paneel Armature Bones (muutmiskontekst Editing, F9)

Hotkey: ⇧ ShiftW, Ctrl⇧ ShiftW, AltW

Menu: Armature » Bone Settings » ... (skelett -> luu seaded)



Skeletiluude paneel
Armature Bones
muutmisrežimis.

Enamik luude seadeid (välja arvatud teisendused) on grupeeritud skeletiluude paneelis Armature Bones igale luule vastavasse alampaneeli (muutmiskontekstis Editing, F9). Toome need siinkohal kõik välja.

Pane tähele, et mõned neist on kättesaadavad ka 3D-vaate hüpikmenüüdest Toggle Setting (⇧ ShiftW või Armature » Bone Settings » Toggle a Setting (skelett -> luu seaded -> seade sisse/välja)), Enable Setting (lülita seade sisse) (Ctrl⇧ ShiftW või Armature » Bone Settings » Enable a Setting (skelett -> luu seaded -> seade sisse)) ning Disable Setting (lülita seade välja) (AltW või Armature » Bone Settings » Disable a Setting (skelett -> luu seaded -> seade välja)) – kõigil kolmel on samad väljad ja nende mõju peaks olema arusaadav.

BO (nimi)

Luu nimi, vaata [allpool](#).

Child of (alluv)

Need kaks seadet määravad luu positsiooni hierarhias, nagu selgitatakse [allpool](#).

Segm (segmente)

Määrab luusegmentide arvu, vaata [allpool](#).

Dist (kaugus), Weight (kaal), Deform (moonuta) (ka ⇧ ShiftW » Deform jne), Mult (mitu) (ka ⇧ ShiftW » Mult VG jne.)

Need seaded (koos luu otspunktide raadiustega) määravad, kuidas luu mõjutab geomeetriat. Seda kirjeldatakse lähemalt [nahaga katmise peatükis](#).

Hinge (hing) (ka ⇧ ShiftW » Hinge jne), S (ka ⇧ ShiftW » No Scale (skaala puudub) jne)

Need määravad alluvate luude käitumise, kui nende ülemaat poseerimisrežimis teisendada; seda kirjeldatakse lähemalt [poseerimise osas](#)!

Hide (peida)

See peidab 3D-vaates luu (sama mis kiirklahv H – loe lähemalt [sellelt leheküljelt](#)).

Lock (lukusta) (ka ⇧ ShiftW » Locked (lukustatud) jne)

See takistab luu redigeerimist muutmisrežimis – vaata lähemalt [eelmiselt lehelt](#).

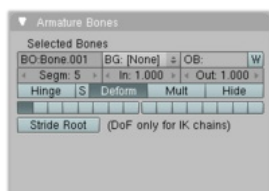
Layers (kihtide nupud)

Nende nuppude abil saab määrata, millistele kihtidele luu kuulub – loe lähemalt [sellest peatükist](#).

Luu jäikuse seaded

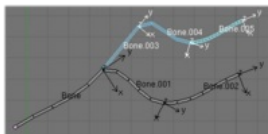
Mode: Muutmisrežiim Edit ja poseerimisrežiim pPose

Panel: Skeletiluude paneel Armature Bones (muutmiskontekst Editing, F9)



Skeletilude paneel
Armature Bones paneel
poseerimisrežiimis.

Ehkki segmentide valik Segm on kättesaadav ka muutmisrežiimis (luude alampaneel skeletilude paneelil Armature Bones), peaksid sa nende "sujuvate" luu omaduste määramiseks minema poseerimisrežiimi (Ctrl+ Tab). Üks põhjus selle kummalise nõude jaoks on see, et muutmisrežiimis joonistatakse luud alati – isegi siis, kui esitusviisiks on B-luu (B-Bone) – pulkadena, nii et sa ei näe selle seade mõju.

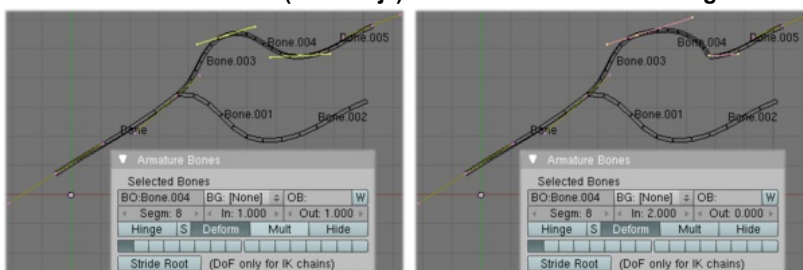


Skelett poseerimisrežiimis,
B-luude esitusviis: luul
Bone.003 on üks segment,
luul Bone.004 neli ja luul
Bone.005 16.

[Eespool](#) me nägime, et luud koosnevad väikestest jäikadest segmentidest, mis joondatakse "virtuaalsele" Bézier' kõverale. Segmentide väli Segm võimaldab luu jaoks määrata nende segmentide arvu – vaikimisi on see **1**, mis tekitab tavalise jäiga luu! Kõrgem väärtus (maksimaalselt **32**) teeb luu painde sujuvamaks, kuid kulutab rohkem aega arvutustele.

Iga luu otspunkt on kõvera "automaatne" juhtpunkt. Sa ei saa muuta juhtpunktide suunda, kuid võid mõjutada nende "pikkust", kasutades välju In (sisse) ja Out (välja), mis vastavalt määravad luu "juurmise" ja "tipmise" juhtpunkti kauguse ahela otsast. Need väärtused on suhtelised vaikepikkuse suhtes, mis loomulikult varieerub automaatselt, sõltudes luu pikkusest, nurgast eelmise/järgmise luuga ja nii edasi.

Luu seadete In/Out (sisse/välja) näide nähtava Bézier' kõveraga.



Vaata luud Bone.004: sellel on vaikimisi In ja Luul Bone.004 on In väärtuseks **2.0** ja Out väärtuseks **0.0**.

Ahela muutmine

Mode: Muutmisrežiim Edit

Panel: Skeletilude paneel Armature Bones (muutmiskontekst Editing, F9)

Hotkey: CtrlP, AltP

Menu: Armature » Parent » ... (skelett -> ülem...)

Sa saad luude hierarhilisi suhteid määrata (ja sellega luua ning muuta luude ahelaid) nii 3D-vaates kui ka nuppude aknas. Ükskõik kumba meetodit sa eelistad, taandub töö sellele, et otsustada iga luu puhul, kas ta on mõne teise luu alluv, ja kui on, kas siis ka sellega seotud.

Et määrata luude alluvust ja/või seoseid, võid sa:

- 3D-vaates valida luu ja *seejärel* tema tulevase ülema ning vajutada CtrlP (või Armature » Parent » Make Parent... (skelett -> ülem -> tee ülemaks)). Ekraanile ilmuvas väikeses ülemaks tegemise menüüs Make Parent vali Connected (ühenda), kui tahad, et luu oleks oma ülemaga seotud; vastasel juhul vali Keep Offset (säilita nihe). Kui sa oled valinud rohkem kui ühe luu, allutatakse nad kõik viimati valitule. Kui sa oled valinud ainult ühe juba allutatud luu või on kõigil valitud luudel juba ülem, on ainuke võimalik tegevus luude sidumine – kui nad juba ei ole seotud. Kui valitud on ainult üks allutamata luu, ilmub veateade Need selected bone(s) (on vaja valitud luid).

Selle meetodi puhul värskest ülemale allutatud luude mõõtkava ega pööret ei muudeta – juhul kui sa sidusid nad ülemaga, nihutatakse nad ainult ülema tipu kohale.

- Sa saad iga valitud luu jaoks määrata ülema nuppude akna skeletilude paneeli Armature Bones vastava luu alampaneeli ülemises paremas nurgas olevast ülemate rippnimekirjast Parent. Kui sa tahad luid siduda, lülita väike seostamise nupp Con nimekirjast paremal sisse.

Selle meetodiga alluva luu tippu ei liigutata – seega, kui seostamise valik Con on sisse lülitatud, siis alluva luu kuju muutub täielikult.

Hierahia määramise näide



Algne skelett, kus luu Bone.005 on allutatud ja seotud luuga Bone.004.



Luu Bone.005 on tõstetud ümber luu Bone.002 alluvaks, kuid ei ole sellega seotud (tulemus on ühesugune, ükskõik kas kasutada kiirvalikut CtrlP2 3D-vaates või skeletiluude paneeli Armature Bones paneeli seadeid).



Luu Bone.005 on allutatud ja seotud luuga Bone.002, kasutades 3D-vaates kiirvalikut CtrlP1.



Luu Bone.005 on allutatud ja seotud luuga Bone.002, kasutades ülemat rippmenüüd Parent luu Bone.005 alampaneelil.

Et luud lahti ühendada või hierahiast eemaldada, võid sa:

- Valida soovitud luud 3D-vaates ja vajutada AltP (või Armature » Parent » Clear Parent... (skelett -> ülem -> eemalda ülem...)). Ilmuvas väikeses ülema eemaldamise menüüs Clear Parent vali Clear Parent (eemalda ülem), et vabastada luu täielikult hierahiast, või Disconnect Bone (ühenda luu lahti), kui sa tahad luud ainult lahti siduda.
- Luude täielikult hierahiast vabastamiseks vali nuppude akna skeletiluude paneelil Armature Bones iga valitud luu puhul tema alampaneelil ülema rippnimekirjast Parent rida "no parent" (ülem puudub). Kui sa tahad lihtsalt luud ülemast lahti siduda, lülita seostamise nupp Con välja.

Pane tähele, et suhted valimata alluvate luudega jäävad muutmata.

Luude nimetamine

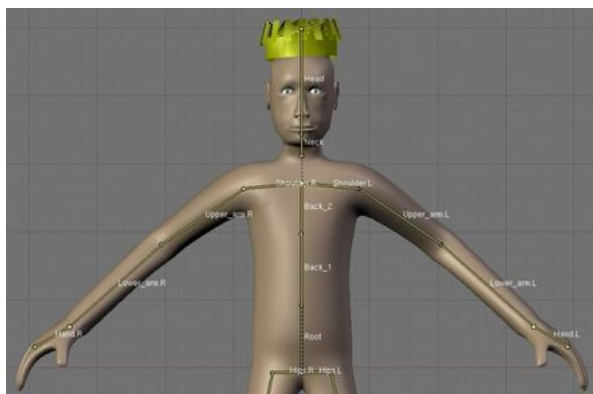
Mode: Muutmisrežiim Edit

Panel: Skeletiluud Armature Bones (muutmiskontekstis Editing, F9), teisendusomadused Transform Properties (3D-vaade, N)

Sa saad luud ümber nimetada kas luu väljal Bone 3D-vaate teisendusomaduste paneelil Transform Properties (N, ainult aktiivset luud) või skeletiluude paneeli Armature Bones vastava luu alampaneeli väljal BO (muutmiskontekstis Editing, F9).

Blenderis on mõned tööriistad, mis tunnevad automaatselt ära õigesti nimetatud (left/right koodidega) sümmeetrilised luud, ja teised, mis võimaldavad luude nimetamist automatiseerida. Vaatame seda detailselt.

Nimetamisreeglid



Näide parema/vasaku (left/right) luude nimedest lihtsas skeletis.

Nimetamisreeglid ei ole vajalikud mitte ainult selleks, et sa õige luu üles leiaksid, vaid annavad ka Blenderile teada, kas antud luul on paariline.

Juhul kui skeletil on sümmeetrilised pooled (st ta on bilateraalsümmeetriline), tasub parema/vasaku nimetamisreeglit kinni pidada. See võimaldab kasutada mõningaid tööriistu, mis säästavad oluliselt aega (nagu X-telje peegelduse tööriist X-Axis Mirror, mida me eespool vaatlesime).

- Kõigepealt tuleb anda luudele tähenduslikud baasnimed. Näiteks *jalg*, *kasivars*, *sorm*, *selg*, *jalalabajne*.
- Kui sul on luu, millest on teisel küljel koopia (paariline), näiteks käsi, lisa talle eraldajaga ees- või lõppliide:
 - Parema/vasaku poole nime eraldaja võib olla kas teises positsioonis (*L_reieluu*) või eelviimasel positsioonis (*reieluu.R*)
 - Kui eraldaja ees või järel on väike- või suurtähtedega *L*, *R*, *left* või *right*, käsitleb Blender seda luud õigesti. Vaata võimalike eraldajate nimekirja allpool. Vali neist üks ja jää taageldamise juures nii püsivalt kui võimalik selle juurde – see tasub ära. Näiteks:

Eraldaja	Võimalikud eraldajad'	näide
(puudub)	kasiLeft	→ kasiRight
_ (alajoon)	Kasi_L	→ Kasi_R
. (punkt)	kasi.I	→ kasi.r

- (sidekriips)	Jalg- l	→ Jalg- r
(tühik)	istmik LEFT	→ istmik RIGHT

Pane tähele, et kõik ülaltoodud näited töötavad ka siis, kui parema/vasaku tähis on luu nime alguses. Lühikest tähist **L/R** saab kasutada ainult koos eraldajaga (st **kasiL/kasiR** ei tööta!).

- Enne kui Blender hakkab skeletti peegeldama, eemaldab ta kõigepealt numbrilise lõpliiite, kui see on olemas (nagu **.001**)
- Sa võid luua koopja luust nimega **bla.L** ja vahetada selle nime, valides **W » Flip Left-Right Names** (vaheta vasaku-parema nimi). Blender paneb koopiale nimeks **bla.L.001** ja nime ümber pööramine annab uueks nimeks **bla.R**.

Luu nime ümber pööramine

Mode: Muutmisrežiim Edit

Hotkey: **W4**

Menu: Armature » Flip Left & Right Names (skelett -> vaheta vasaku ja parema nimi)

Sa saad valitud luude nimedes parema ja vasaku poole koodid (vaata eestpoolt) vahetada kas menüüst Armature » Flip Left & Right Names ((skelett -> vaheta vasaku ja parema nimi) või Specials » Flip Left-Right Names (lisavõimalused -> vaheta vasaku ja parema nimi) (**W4**). See võib olla kasulik, kui sa oled loonud poole sümmeetrilisest skeletist (kus luud on nimetatud parema või vasaku poole koodidega), duplitseerinud ja peegeldanud selle ja tahad nimesid teise poole jaoks korrigeerida. Blender vahetab luunimede tekstis koodid vastavalt ülaltoodud nimetamisreeglitele, eemaldades nime lõpust võimaluse korral numbrid.

Automaatne luude nimetamine

Mode: Muutmisrežiim Edit

Hotkey: **W5, W6, W7**

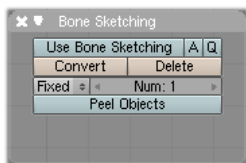
Menu: Armature » AutoName Left-Right (skelett -> automaatne vasaku-parema nimetamine), Armature » AutoName Front-Back (skelett -> automaatne esimese-tagumise nimetamine), Armature » AutoName Top-Bottom (skelett -> automaatne ülemise-alumise nimetamine)

Kolm automaatse nimetamise valikut AutoName skeleti menüüs Armature ja lisavõimaluste menüüs Specials (**W**) võimaldavad sul automaatselt lisada valitud luude nimedele lõpliiited, *vastavalt nende juurte asendile skeleti keskme ja skeleti koordinaatsüsteemi suhtes*:

- AutoName Left-Right (automaatne vasaku-parema nimetamine) lisab lõpu **.L** kõigi luude nimedele, *mille juure X-koordinaat on positiivne*, ja lõpu **.R** kõigi luude nimedele, *mille juure X-koordinaat on negatiivne*. Kui juur asub täpselt X-telje punktis **0.0**, kasutatakse luu tüüpi X-koordinaati. Kui luu mõlemad otspunktid on X-telje punktis **0.0**, lisatakse luu nimele punkt, aga mitte liidet **L** ega **R** (sest Blender ei suuda otsustada, kas see on vasaku või parema poole luu).
- AutoName Front-Back (automaatne esimese-tagumise nimetamine) lisab lõpu **.Bk** (taga) kõigi luude nimedele, *mille juure Y-koordinaat on positiivne*, ja lõpu **.Fr** (ees) kõigi luude nimedele, *mille juure Y-koordinaat on negatiivne*. Juhul kui Y-koordinaat on **0.0**, kehtib sama skeem, mis automaatsel vasaku-parema nimetamisel.
- Automaatne ülemise-alumise nimetamine (AutoName Left-Right) lisab lõpu **.Top** (üleval) kõigi luude nimedele, *mille juure Z-koordinaat on positiivne*, ja lõpu **.Bot** (all) kõigi luude nimedele, *mille juure Z-koordinaat on negatiivne*. Juhul kui Z-koordinaat on **0.0**, kehtib sama skeem, mis automaatsel vasaku-parema nimetamisel.

Luude skitseerimine (*Bone Sketching*)

Kui sa arvad, et terve taagelduse tegemine luuhaaval on igav, siis ole õnnelik: mõnedel Blenderi arendajatel oli sama tunne ning nad löid skeleti skitseerimise tööriista (*Skeleton Sketching*), mida varem tunti Etch-a-ton'i nime all ja mis lubab sul korraga "joonistada" (skitseerida) terveid luude ahelaid.



Luude skitseerimise paneel Bone Sketching oma algses (mitteaktiivses) seisus.

Etch-a-ton on loomulikult kasutatav ainult 3D-vaates muutmisrežiimis Edit. Seda kasutatakse läbi luude skitseerimise paneeli Bone Sketching (P või menüü *Armature » Bone Sketching*) ning hiire abil (joonte tõmbamiseks LMB ja žestide jaoks RMB). Tööriistapaneeli käivitamine ei luba veel skitseerimist – sa pead luude ahelate joonistamiseks vajutama nuppu *Use Bone Sketching* (vastasel juhul jääd sa tavapärasesse muutmisrežiimi Edit).

Skitseeritakse kahes osas:

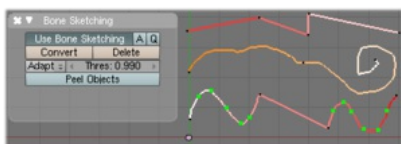
1. [Joonistades mõned "sujuvad" ja/või polügonaalised jooned](#) (mida kutsutakse "tõmmeteks" (*strokes*)). Iga tõmme vastab ühele luude ahelale.
2. [Konverteerides need tõmbed reaalseks luude ahelateks](#) (kasutades erinevaid meetodeid).

Vaatepunkt on oluline, sest see määrab tulevikus luude pöördenu: *valmiva luu Z-telg joondatakse vastavalt valitud 3D-vaate Z-teljele, kus sa tõmbe joonistad (kui sa just ei kasuta malli alusel konverteerimise meetodit Template)*. Tõmbed joonistatakse valitud vaatepinnale, nii et nad läbiksid 3D-kursori, kuid sa saad tekitada mingil määral "kolmemõõtmelisi" tõmbeid, kasutades erinevates vaadetes parandamise võimalust *Adjust* (vaata [altpoolt](#)).

Kui sa lülitad sisse kiire skitseerimise võimaluse *Quick Sketch*, liidetakse need kaks sammu üheks: pärast tõmbe lõpetamist (vaata [altpoolt](#)) konverteeritakse see koheselt luuks (kasutades parajasti aktiivset meetodit) ja kustutatakse. See võimalus muudab luude skitseerimise kiireks ja tõhusaks, kuid sa kaotad keerulisemad tõmbe muutmise viisid.

Eskiise Blenderi failidesse ei salvestata ja seega ei saa sa skitseerimist ilma oma tööd kaotamata katkestada! Samuti pane tähele, et *skitseerimine on terves Blenderis ühine*, st eksisteerib ainult üks tõmmete komplekt (üks eskiis), mitte üks iga skeleti või isegi faili kohta eraldi.

Ahelate joonistamine



Tõmmete näide. Ülevaalt alla:

*Valitud neljast sirgest segmendist koosnev polügonaalne tõmme, mis kulgeb vasakult paremale.

*Valimata kahest segmendist koosnev vabakäetõmme, mis kulgeb vasakult paremale.

*Valitud segatõmme, milles on üks sirge tõmme kahe vaba vahel ja mis kulgeb vasakult paremale.

Pane samuti tähele, et muudetav skelett on tühi (selles pole luid) ning seetõttu näed sa selle objekti keset (all ääres olev roosa punkt).

Niisiis, iga tõmme, mille sa teed, on üks luude ahel, mis kulgeb esimesest punktist (punaseimast või kõige oranžimast tõmbe osast) lõpuni (kõige valgem osa). Tõmme koosneb mitmest *segmendist*, mille vahekohti tähistavad väikesed mustad täpid – *alati eksisteerib vähemalt üks luu segment* (selle erandiks on malli järgi konverteerimise meetod *Template*; vaata [järgmist lehekülge](#)) ning seetõttu tähistavad mustad punktid valmivate luude otsi. Segmente on kahte sorti ja neid saab omavahel segada:

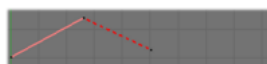
Sirged segmendid

Sirge segmendi loomiseks vajuta selle algpunktis klahvi {Shortcut|lmb}}. Seejärel liiguta ilma ühtegi nuppu vajutamata hiirekursorit: tulevast luud tähistab punane punktiirjoon. Vajuta uuesti klahvi LMB, et segmenti lõpetada.

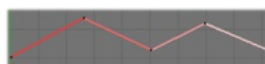
Iga tõmbe sirge segment loob alati ainult ühe luu, hoolimata sellest, millist konverteerimise algoritmi sa kasutad (välja arvatud malli järgi konverteerimise meetod *Template*).



Esimene segment on alustatud (klahvi LMB vajutus) ja hiir on liigutatud



Esimene segment on teise klahvi LMB vajutusega lõpetatud ning see alustas



Nende sammude kordamisel on meil nüüd neljast segmendist koosnev

segmenti lõpp-punkti omakord ka teist segmenti polügonaalne tõmme.
Sirgete segmentide joonistamise näide.

Vabakäe segmentid

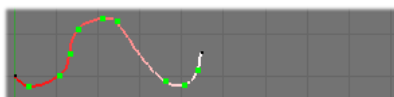
Et luua segmente vaba käega (köverduvaid), vajuta alguspunktis klahvi LMB ja hoia seda all. Seejärel joonista oma segment hiirekursorit liigutades – nagu igas teiseski joonistamisprogrammis! Segmenti lõpetamiseks lase klahv LMB lahti – seejärel hakkaksid sa looma uut *sirget* segmenti, nii et kui soovid hoopis alustada uut *vabakäe* segmenti, pead klahvi LMB koheselt uuesti alla vajutama.

Tõmbe vabakäe segmentid tekitavad erinevate meetodite järgi erineva arvu luid (sõltuvalt valitud konverteerimismeetodist). Tulevaste luude parajasti valitud meetodist tingitud lõpp-punkte tähistavad segmentide peal olevad väikesed rohelised punktid, mida näidatakse *ainult parajasti valitud tõmbe peal*.

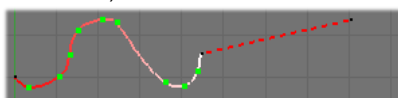
Vabakäe segmentidega joonistamine kasutab sama Manhattani kauguse sätet Manhattan Dist nagu [rasvakriidi tööriist](#) (kasutaja eelistuste aken User Preferences, muutmismeetodite paneel Edit Methods, rasvakriidi grupp Grease Pencil), et määrata, millal segmentile uus punkt lisada – seega, kui sinu arust on vabakäe segmentid liiga detailised, tõsta natuke selle sätte väärtust, ning kui leiad, et need on liiga rohkemad, vähenda väärtust.



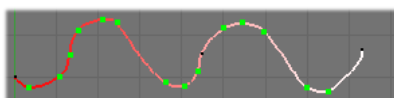
Esimese vabakäe segmenti joonistamise ajal (klahvi LMB all hoides hiire lohistamine).



Esimese vabakäe segmenti lõpetamine (klahv LMB lahti lastud).



Kui sa nüüd liigutad hiirt ilma klahvi LMB uuesti all hoidmata, lood sa sirge segmenti



Kuid kui sa koheselt vajutad klahvi LMB uuesti alla ja asud lohistama, alustad sa hoopis uut vabakäe segmenti.

Vabakäe segmenti joonistamise näidis.

Terve tõmbe lõpetamiseks vajuta klahvi RMB. Tühista tõmbe joonistamine, vajutades klahvi Esc. Sa saad tõmbeid ka alloleva võrega nakkuma panna, hoides joonistamise ajal all klahvi Ctrl. Muideks, luude skitseerimise paneeli Bone Sketching allotsas olev objekti koorimise nupp Peel Objects on "sama" kui nakkumise päise sätete all olev "pärldiku" nupp, mis on nähtav siis, kui valitud nakkumise elemendiks on maht (Volume) – rohkem infot [võrega nakkumise](#) lehel.

Tõmmete valimine

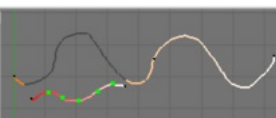
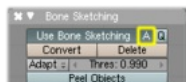
Tõmme saab olla valitud (kujutatakse punasest valgeks mineva joonena) või valimata (kujutatakse oranžist valgeks mineva joonena) – vaata ülalolevat pilti (*Tõmmete näide*). Nagu tavaliselt, saad tõmme valida vajutades sellel klahvi RMB, lisada või eemaldada praegusest valikust klahvikombinatsiooniga \diamond Shift RMB ning kõiki tõmberid valida/valikut tühistada klahviga A...

Kustutamine

Klahv X või kustutamispupu Delete vajutamine (luude skitseerimise paneelis Bone Sketching) kustutab valitud tõmbed (ole ettevaatlik, sest siinkohal ei näidata ühtegi hoiatavat või kinnitavat hüpikmenüüd...). Vaata ka [allpool olevat žestide kirjeldust](#).

Tõmmete muutmine

Sa saad oma tõmbeid parandada või "uuesti joonistada", lülitades sisse luude skitseerimise paneelis Bone Sketching oleva ülejoonistamise võimaluse Overdraw Sketching. See muudab tõmmete joonistamise käitumist (st klahvi LMB vajutused ja/või allhoidmised): kui sa joonistad, ei teki sa uut tõmme, vaid muudad hoopis hiirele lähimat tõmme. Vana tõmbe osa, mis kuulub asendamisele, kuvatakse hallina. See võimalus ei arvesta tõmmete valikuga, st sel moel saab muuta kõiki tõmbeid ja mitte ainult valitud. Samuti pane tähele, et selle võimaluse sisselülitamisel ei muuda sa ühtegi varem olemasolevat joont, kui su hiir on neist liiga kaugel, vaid tekitab hoopis uue joone, justkui valik Overdraw Sketching oleks välja lülitatud.



Tõmbe parandamine: "valimata" (oranži) tõmbe hall osa asendatakse parajasti joonistatud "asendustõmbega".

Parandatud tõmme.

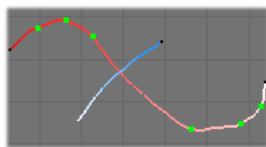
Tõmbe parandamise näide.

Lõpetuseks pea meeles, et skitseerimise puhul ei eksisteeri ennistamise/uuesti tegemise võimalust.

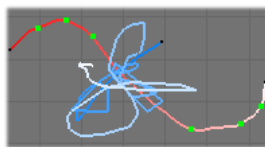
Žestid

Päris paljud tõmmete võimalused on kasutatavad ainult žestide (*gestures*) vahendusel. Žestide alustamiseks vajutatakse alla klahvid \diamond Shift+LMB (kui sa parajasti juba ei joonista mõnda tõmme) ning neid kujutatakse sinisest valgeks minevate joontena. Žest võib mõjutada korraga mitut tõmme.

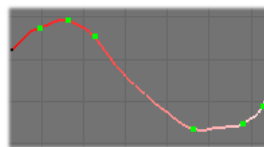
Pärast žesti "tegemise" alustamist ei ole otsest meetodit, kuidas seda tühistada. Seega on parim lahendus, kui oled oma meelt muutnud (või teinud "vale liigutuse"), žesti lihtsalt jätkata, kuni tulemuseks on jube soperdis, mis ületab mitmel korral iseenda joont – ehk siis lühidalt miski, mida žestide süsteem mitte kunagi ära ei tunneks!



Põrgu! Ma ei tahtnud seda tõmmet siit kohalt lõigata!



Soperdame pisut...

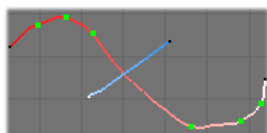


Vedas! See läks napilt, kuid see tõmme on ikka veel ühes tükis...

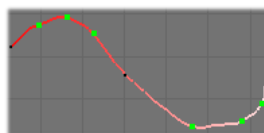
Žesti tühistamise näide.

Lõikamine

Segmendi **lõikamiseks** (st lisa sellesse uus must punkt, tehes ühest segmendist kaks), "joonista" valitud joonest soovitava lõike kohal üle sirge joon.



Žest.
Lõikežesti näidis.



Tulemus.

Kustuta

Tõmbe **kustutamiseks** joonista tõmmet kaks korda ületav "V".



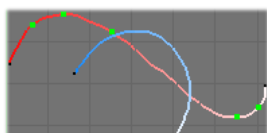
Žest.
Kustutamisžesti näidis.



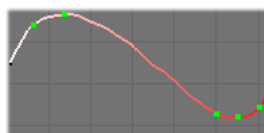
Tulemus.

Pööra tagurpidi

Tõmbe **tagurpidi pööramiseks** (st et tulevased ahela luud oleks vastupidises järjekorras) joonista tõmmet kaks korda ületav "C".



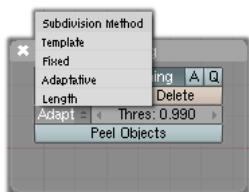
Žest.
Tagurpidi pööramise žesti näidis.



Tulemus.

Luudeks konverteerimine

Kui sul on üks või rohkem valitud tõmmet, saad need luudeks konverteerida, kasutades kas luude skitseerimise paneelis Bone Sketching olevat konverteerimise nuppu Convert või vastavat žesti (loe [üla](#)lt). Iga valitud tõmme tekitab ühe luude ahela, mille suund on kõige punasemast kõige valgemaks. Pane tähele, et tõmbe konverteerimine ei kustuta seda.



Luude skitseerimise paneelis Bone Sketching oleva tükeldamise meetodi rippnimekiri Subdivision Method.

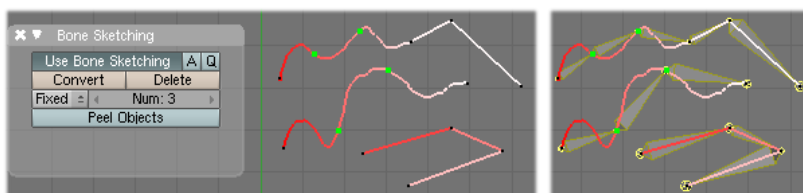
On neli erinevat konverteerimise meetodit – kolm "lihtsat" ning üks keerulisem ja võimsam (Template), mis kasutab samas skeletis või mõnes teises olevaid luid uuesti mallidena, konverteeritavate tõmmete jaoks ning mida kirjeldatakse [järgmisel leheküljel](#). Igal juhul pea meeles, et *sirged segmendid konverteeritakse alati ainult üheks luuks* (välja arvatud malli järgi konverteerimise meetodi

Template puhul) ning et valitud vabakäe segmentidel näidatakse tulevaste luude lõppe roheliste punktidenä.

Samuti pea meeles, et tekkivate luude pööre määratakse tõmbe joonistamise ajal (välja arvatud malli järgi konverteerimise meetodi Template puhul) – nende Z-telg joondub joonistamise ajal aktiivse 3D-vaate Z-teljega.

Fikseeritud (*Fixed*)

Selle meetodi puhul jagatakse valitud tõmmete iga vaba käe segment ühtlaselt n osaks (mille arv määratakse numbriväljaga Num), st tulemuseks on n arv luid.



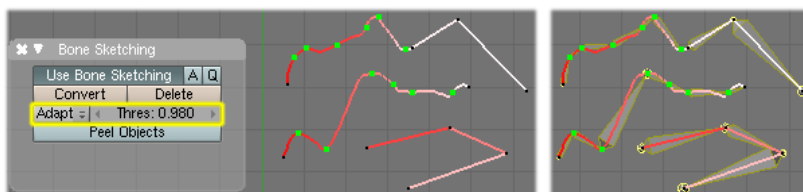
Fikseeritud konverteerimise (Fixed) sätted ja nende eelvaade valitud tõmmetel.

Fikseeritud konverteerimise tulemus.

Tõmbe fikseeritud meetodil luuks konverteerimise tulemus.

Kohalduv (*Adaptive*)

Selle meetodi puhul loob iga valitud tõmbe vabakäe segment nii palju luid kui vajalik, et järgida võimalikult täpselt oma kuju – selle "võimalikult täpse" järgimise parameetri määrab läve väli Thres (kõrgemate väärtuste tulemuseks on suurem arv luid, mistõttu jälgitakse segmenti kuju täpsemalt). Seega, mida kõveraterohkem on segment, seda enam luid luuakse.



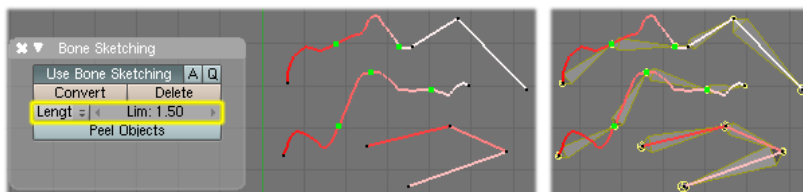
Kohalduva konverteerimise (Adaptive) sätted ja nende eelvaade valitud tõmmetel.

Kohalduva konverteerimise tulemus.

Tõmbe kohandatud meetodil luuks konverteerimise tulemus.

Pikkus (*Length*)

Selle meetodiga loob iga valitud tõmbe vabakäe segment niipalju luid kui vaja, nii et üksik neist ei oleks pikem kui pikkuse välja Len väärtus (Blender ühikutes).



Pikkuse järgi konverteerimise (Adaptive) sätted ja nende eelvaade valitud tõmmetel.

Pikkuse järgi konverteerimise tulemus.

Tõmbe pikkusel põhineval meetodil luuks konverteerimise tulemus.

Mall (*Template*)

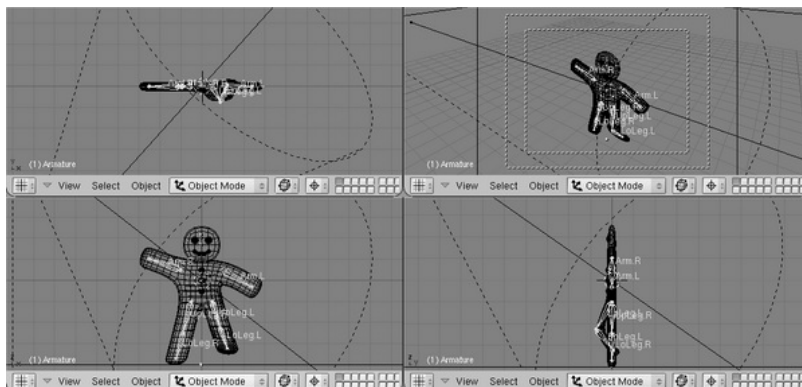
See on keeruline teema, mida kirjeldatakse [eraldi leheküljel](#).

Sihtpunkti uuesti märkimine (*Retarget*)

...

Nahaga katmine (*Skinning*)

[Eelmistel lehekülgedel](#) me nägime, kuidas luua skeletti, teha luude ahelaid jne. Kena skelett ei ole aga meie eesmärk omaette – kui sa just ei soovi luua “Surmatantsu” animatsiooni, tahad sa ilmselt oma skeletile lihased ja naha lisada! Skeleti “sidumist või linkimist” objektidega, mida ta hakkab teisendama või deformeerima, nimetatakse “nahaga katmiseks” (*Skinning*).



Skeletiga seotud piparkoogimehike

Blenderis on kaks põhilist nahaga katmise viisi:

- Sa võid seada [luudele ülem-/piirangobjektid](#) – kui sa seejärel teisendad luid poseerimisrežiimis Pose, teisendatakse ka nende “alluvaid”, täpselt nii, nagu võib eeldada hierarhilises suhtes objektidelt. *Selle meetodiga aga “alluvad” ei deformeeru mitte kunagi.*
- Sa võid kasutada [tervel võrel skeletitöötlejat \(Armature Modifier\)](#) ja seejärel seostada üksikud võre osad üksikute luudega selles skeletis. See on palju keerukam, aga ka võimsam meetod ja *ainus võimalus objekti geomeetria moonutamiseks*, st tema tippude/juhtpunktide suhteliste asukohtade muutmiseks.

Poseerimine (*Posing*)

Kui sa oled oma skeleti kõigi vajalike objektide abil [nahaga katnud](#) (*skin*), saad sa hakata seda poseerima. Sisuliselt tähendab see seda, et luid teisendades moondad või teisendada sa ka nahaobjekte. Seda ei tee sa aga muutmisrežimis Edit – tuleta meelde, et selles režimis muudad sa *oma skeleti vaike-, alus- ehk "puhkepoosi"*. Samuti ei saa sa kasutada objektirežiimi Object, sest seal saad sa teisendada objekte ainult tervikuna.

Seepärast on Blenderis skeleti poseerimiseks olemas kolmas töörežiim - poseerimisrežiim Pose. See on sisuliselt "objektirežiim üksikute luude jaoks". Puhkepoosis (nagu on määratud muutmisrežimis) on iga luu asendil/pöördel ja skaalal neutraalsed väärtused (st asendi ja pöörde jaoks **0.0** ja skaala jaoks **1.0**). Seega, kui sa muudad luud poseerimisrežimis, annad sa tema teisendustele mingi suhtelise erinevuse võrreldes puhkeasendiga – tõtt-öelda samaneb see tehniliselt võrede [suhtelistele võtmevormidele](#).

Poseerimise osa ülevaade

Selles peatükis me käsitleme:

- Poseerimisrežiimile spetsiifilisi [visualiseerimisvõimalusi](#).
- Kuidas selles režimis luid [valida ja muuta](#).
- Kuidas kasutada [pooside teeki](#) (*Pose Library*).
- Kuidas määrata [piirajate \(Constraints\) abil](#) luude võimalikke vabadusastmeid.
- Kuidas kasutada [pöördkinemaatikat](#).

Kuigi seda võib kasutada ka staatiliste stseenide jaoks, on poseerimine tihedalt seotud [animatsioonitehnikatega](#). Selles osas püüame me põhiliselt käsitleda animatsioonist sõltumatut poseerimist, ehkki see alati ei õnnestu. Seega, kui sa ei tea veel midagi Blenderi animatsioonidest, on võib-olla hea idee lugeda enne vastavat peatükki (vaata linki ülalpool) ja siis tulla tagasi selle osa juurde.

Vaata ka

Nagu ikka, vaata ka [õppetükke](#), kust leiab praktilisemaid tegevusjuhiseid, ja eriti [seda BSoD juhendit](#).

Pöördkinemaatika (*Inverse Kinematics*)

Pöördkinemaatika (*Inverse Kinematics*, IK) on animatsiooni loomist lihtsustav meetod, mis võimaldab teha palju täiuslikumaid animatsioone vähema vaevaga.

Automaatne IK (*Automatic IK*)

Automaatne IK on tööriist lihtsate demoposide tegemiseks. See asub 3D-vaate tööriistariiulil. Seda tasub kasutada ainult kiirete ja ebatäpsete näidiste jaoks, sest see ei võimalda kasutada piiranguid. Selle efekt on sama, kui sa määraksid iga luuahela viimasele luule IK piiraja, mille ahela pikkus on 0 (lõpmatus).

IK piirajad (*IK Constraints*)

...

IK splain (*Spline IK*)

IK splain on piiraja, mis paigutab ahela luud piki kõverat. Liites ühelt poolt kõverate poolt pakutava paindlikkuse ja lihtsuse, millega saab nende abil luua kenasid kujundeid, ning teiselt poolt luude poolt pakutava täpse juhtimise ja kontrolli, on IK splain asendamatu tööriist taageldaja tööriistakastis. Ta sobib eriti hästi painduvate kehaosade, nagu sabade, kombitsate ja selgroogude ning elutute objektide, näiteks kõite taageldamiseks.

IK splaini seadistuste täielik kirjeldus on peatükis [IK splain](#).

Kasutamise põhipunktid

IK splain ei ole pedantselt võttes 'pöördkinemaatiline' meetod (st IK piiraja), vaid hoopis 'otsekinemaatiline' meetod (st tavaline luu poseerimine). Siiski on tal mitmed IK piirajate omadused, näiteks töötab ta korraga mitmel luul, teda ei saa kasutada objektidel ja ta arvutatakse välja alles pärast kõigi teiste piirajate arvutamist. Tuleb tähele panna, et kui luud mõjutavad korraga nii tavaline IK ahel kui ka IK splaini ahel, siis jääb peale tavaline IK ahel. Selliseid olukordi tuleks aga pigem vältida, sest nende tulemusi võib olla keerukas juhtida.

IK splaini loomiseks peab sul olema omavahel seotud luude ahel ja kõver, mis hakkab nende luude asendit piirama.

1. Olles valinud ahela viimase luu, lisa sellele omaduste akna luupiirajate kontekstis (*Bone Constraints*) piiraja [Spline IK](#) (IK splain).
2. Määra ahela pikkuse valiku *Chain Length* väärtuseks luude arv ahelas, mida kõver hakkab mõjutama (kaasa arvatud valitud luu).
3. Lõpuks määra sihtmärgi väljal *Target* kõver, mis hakkab luude ahelat juhtima.

Palju õnne! Luude ahel järgib nüüd kõverat.

Parameetrid ja seaded

Pööre (*Roll*)

IK splaini 'pöörlemise' või 'keerdumise' juhtimiseks saab kasutada tavalist luude pööramist ümber nende Y-telgede. Näiteks kui tahad pöörata ahela kõiki luid mingist punktist edasi, pööra lihtsalt seda luud ümber Y-telje. Pöörde kopeerimise piiraja määramine peaks samuti töötama.

Nihe (*Offset*)

Kui ahela nihke valik *Chain Offset* on sisse lülitatud, siis saab kõvera kuju järgivat ahelat vabalt suvalisse ruumpunkti ümber paigutada. Vaikimisi on see valik sisse lülitatud ja luude ahel järgib kõverat ilma teisendust arvestamata.

Jämedus (*Thickness*)

XZ-skaleerimisrežiimi valikuga *XZ Scale Mode* saab juhtida ahela luude jämedust. Selle abil määratakse meetod, kuidas arvutada ahela iga luu X- ja Z-telgedel toimuvat skaleerimist.

Võimalikud režiimid on:

- Puudub (None) – selle valikuga on X- ja Z-mõõtkava väärtuseks 1.0
- Säilita maht (Volume Preserve) – X- ja Z-mõõtkavade väärtused on Y-mõõtkava (luu pikkuse) pöördväärtused, nii et luu 'ruumala' püsib konstantsena
- Luu esialgne (Bone Original) – kasutab neid X- ja Z-skaalasid, mis oleksid luul tavalise arvutamise meetodi puhul

Lisaks neile režiimidele on olemas veel kõvera raadiuse kasutamise valik *Use Curve Radius*. Kui see valik on sisse lülitatud, kasutatakse luu X- ja Z-mõõtkava leidmiseks kõvera nende punktide, kuhu luu otspunktid langevad, keskmist kõverusraadiust. See võimaldab ülatoodud meetoditel leitud skaalasid parema väljanägemise saamiseks täpsustada.

Nõuanded hea tulemuse saamiseks

- Paremate deformatsioonide saamiseks on soovitatav, et luud oleksid enam-vähem ühepikkused ja mitte liiga pikad, et nad sobituksid kenasti kõverale. Samuti peaks luud ideaalis olema loodud niimoodi, et nad järgiksid kõvera kuju tema 'puhkeasendis', et minimeerida moonutusi kohtades, kus kõver teeb järske pööreid – need tulevad eriti hästi välja siis, kui venitamine on välja lülitatud.
- Kõvera kuju juhtimiseks on soovitatav kasutada haake (eriti Blender 2.5 uusi luuhaake (*Bone Hooks*)), mis mõjuvad kõvera juhtpunktidele, üks haak punkti kohta. Üldiselt piisab kõvera jaoks vähestest juhtpunktidest (st 1 punkt iga 3–5 luu kohta annab piisavalt paindlikkust).

- Kõvera tüübil ei ole tähtsust – juhul kui temast on võimalik teha rada, mida saab kasutada rajapiirajaga (*FollowPath Constraint*), töötab ta ka IK splainiga. See sõltub olulisel määral sellest, kui täpselt on tema kuju vaja haakidega juhtida.
- Skelettide loomisel on hetkel vajalik, et kõvera kuju määravad luud asuksid teises skeletis kui võre kuju muutvad luud (st need deformeerivad luud, millele IK splain on rakendatud). See on nii selleks, et vältida näivaid "sõltuvustsükleid", sest Blenderi sõltuvusgraafik suudab juhtluude, kõverate ja IK splainiga luude omavahelisi sõltuvusi analüüsida ainult objekti kaupa.

IK splain (*Spline IK*)

IK splain on piiraja, mis paigutab ahela luud piki kõverat. Liites ühelt poolt kõverate poolt pakutava paindlikkuse ja lihtsuse, millega saab nende abil luua kenasid kujundeid, ning teiselt poolt luude poolt pakutava täpse juhtimise ja kontrolli, on IK splain asendamatu tööriist taageldaja tööriistakastis. Ta sobib eriti hästi painduvate kehaosade, nagu sabade, kombitsate ja selgroogude ning elutute objektide, näiteks kõite taageldamiseks.

IK splaini seadistuste täielik kirjeldus on peatükis [IK splain](#).

Kasutamise põhipunktid

IK splain ei ole pedantselt võttes 'pöördkinemaatiline' meetod (st IK piiraja), vaid hoopis 'otsekinemaatiline' meetod (st tavaline luu poseerimine). Siiski on tal mitmed IK piirajate omadused, näiteks töötab ta korraga mitmel luul, teda ei saa kasutada objektidel ja ta arvutatakse välja alles pärast kõigi teiste piirajate arvutamist. Tuleb tähele panna, et kui luud mõjutavad korraga nii tavaline IK ahel kui ka IK splaini ahel, siis jääb peale tavaline IK ahel. Selliseid olukordi tuleks aga pigem vältida, sest nende tulemusi võib olla keerukas juhtida.

IK splaini loomiseks peab sul olema omavahel seotud luude ahel ja kõver, mis hakkab nende luude asendit piirama.

1. Olles valinud ahela viimase luu, lisa sellele omaduste akna luupiirajate kontekstis (*Bone Constraints*) piiraja [Spline IK](#) (IK splain).
2. Määra ahela pikkuse valiku *Chain Length* väärtuseks luude arv ahelas, mida kõver hakkab mõjutama (kaasa arvatud valitud luu).
3. Lõpuks määra sihtmärgi väljal *Target* kõver, mis hakkab luude ahelat juhtima.

Palju õnne! Luude ahel järgib nüüd kõverat.

Parameetrid ja seaded

Pööre (*Roll*)

IK splaini 'pöörlemise' või 'keerdumise' juhtimiseks saab kasutada tavalist luude pööramist ümber nende Y-telgede. Näiteks kui tahad pöörata ahela kõiki luid mingist punktist edasi, pööra lihtsalt seda luud ümber Y-telje. Pöörde kopeerimise piiraja määramine peaks samuti töötama.

Nihe (*Offset*)

Kui ahela nihke valik *Chain Offset* on sisse lülitatud, siis saab kõvera kuju järgivat ahelat vabalt suvalisse ruumpunkti ümber paigutada. Vaikimisi on see valik sisse lülitatud ja luude ahel järgib kõverat ilma teisendust arvestamata.

Jämedus (*Thickness*)

XZ-mõõtkava teisendamise režiimi valikuga *XZ Scale Mode* saab juhtida ahela luude jämedust. Selle abil määratakse meetod, kuidas arvutada ahela iga luu X- ja Z-telgedel toimuvat mõõtkava teisendamist.

Võimalikud režiimid on:

- Puudub (None) – selle valikuga on X- ja Z-mõõtkava väärtuseks 1.0
- Säilita maht (Volume Preserve) – X- ja Z-mõõtkavade väärtused on Y-mõõtkava (luu pikkuse) pöördväärtused, nii et luu 'ruumala' püsib konstantsena
- Luu esialgne (Bone Original) – kasutab neid X- ja Z-skaalasid, mis oleksid luul tavalise arvutamise meetodi puhul

Lisaks neile režiimidele on olemas veel kõvera raadiuse kasutamise valik *Use Curve Radius*. Kui see valik on sisse lülitatud, kasutatakse luu X- ja Z-mõõtkava leidmiseks kõvera nende punktide, kuhu luu otspunktid langevad, keskmist kõverusraadiust. See võimaldab ülaltoodud meetoditel leitud skaalasid parema väljanägemise saamiseks täpsustada.

Nõuanded hea tulemuse saamiseks

- Paremate deformatsioonide saamiseks on soovitatav, et luud oleksid enam-vähem ühepikkused ja mitte liiga pikad, et nad sobituksid kenasti kõverale. Samuti peaks luud ideaalis olema loodud niimoodi, et nad järgiksid kõvera kuju tema 'puhkeasendis', et minimeerida moonutusi kohtades, kus kõver teeb järske pöörded – need tulevad eriti hästi välja siis, kui venitamine on välja lülitatud.
- Kõvera kuju juhtimiseks on soovitatav kasutada haake (eriti Blender 2.5 uusi luuhaake (*Bone Hooks*)), mis mõjuvad kõvera juhtpunktidele, üks haak punkti kohta. Üldiselt piisab kõvera jaoks vähestest juhtpunktidest (st 1 punkt iga 3–5 luu kohta annab piisavalt paindlikkust).
- Kõvera tüübil ei ole tähtsust – juhul kui temast on võimalik teha rada, mida saab kasutada rajapiirajaga (*FollowPath Constraint*), töötab ta ka IK splainiga. See sõltub olulisel määral sellest, kui täpselt on tema kuju vaja haakidega juhtida.
- Skelettide loomisel on hetkel vajalik, et kõvera kuju määravad luud asuksid teises skeletis kui võre kuju muutvad luud (st need deformeerivad luud, millele IK splain on rakendatud). See on nii selleks, et vältida näivaid "sõltuvustsükleid", sest Blenderi sõltuvusgraafik suudab juhtluude, kõverate ja IK splainiga luude omavahelisi sõltuvusi analüüsida ainult objektikaupa.

Piirajad

Kirjeldus

Piirajad on objekti lisad, mis juhivad erinevatel viisidel objekti teisendusi (asukoht, pööre, mõõtkava). Sisuliselt on piirajad objekti tasemel analoogid [töötajatele](#), mis mõjutavad "objektiandmeid" (näiteks vööred, kõverad jne).

Kõigil piirajatel on [ühine põhikasutajaliides](#), millel on samuti palju sarnasusi töötajate kasutajaliidesega.

Piirajate kasutamine

Kuigi piirajad võivad olla kasulikud ka staatiliste stseenide puhul (näiteks objektide automaatseks paigutamiseks, pööramiseks või mõõtkava teisendamiseks), kavandati nad algselt animatsiooni tarbeks, kuna lubavad piirata/juhtida objekti liikumisvabadust kas absoluutselt (st globaalses koordinaadistikus) või teiste objektide suhtes.

Tasub teada, et sisemiselt töötavad piirajad ainult 4×4 teisendusmaatriksitega. Kui sa kasutad konkreetse pöörde või mõõtkava piiramist, võetakse see informatsioon otse maatriksist, mitte luu ega objekti sätetest. Pöörete kombineerimine ebaühtlase või negatiivse mõõtkava teisendamisega võib anda mitteootuspäraseid tulemusi.

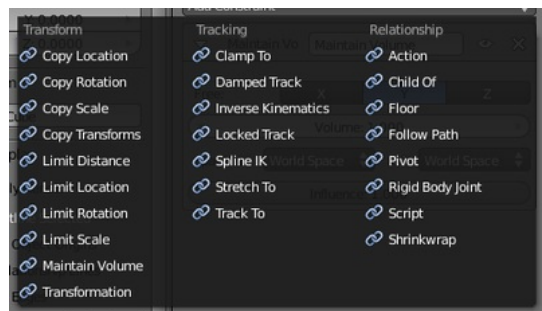
Luude piiramine

Lõpuks ometi on Blenderis suurepärane taageldusviis - poseerimisrežiim Pose, kus iga skeleti luu käitub nagu tavaline objekt ja tema asendile saab rakendada piiranguid. Enamik piirajaid töötab hästi nii objektide kui ka luude puhul. Siiski on erandeid, mis tuuakse välja asjakohastes peatükkides.

Loe lisaks:

- piirajate kasutamisest objektide animeerimisel [animatsiooni peatükist](#)
- piirajate kasutamisest taageldamisel [skelettide peatükist](#)

Olemasolevad piirajad



Constraint (piirajate) menüü

Piirajaid on mitut erinevat tüüpi. Neid võib jagada kolme perekonda:

- [Teisendamise piirajad \(Transform\)](#)
- [Järgimise piirajad \(Tracking\)](#)
- [Suhte piirajad \(Relationship\)](#)

Osad piirajad töötavad ainult *omanikuga*, teised vajavad lisaks teist objekti (sihtmärki), mis mõnel juhul peab olema spetsiifilist tüüpi (näiteks kõver). Sellisel puhul kuvatakse sihtmärgile suunatud piirajad 3D-vaates tumesinise punktiirjoonega valdaja ning sihtmärgi vahel.

Teisendamise piirajad

Need piirajad kontrollivad ja piiravad oma valdaja koordinaatteisenduse omadusi kas absoluutse väärtusena või oma sihtmärgi omaduste suhtes.

Asukoha kopeerija (Copy Location)	Kopeerib sihtmärgi asukoha (valikulise nihkega) valdajale nii, et mõlemad liiguvad koos.
Pöörde kopeerija (Copy Rotation)	Kopeerib sihtmärgi pöörde (valikulise nihkega) valdajale, nii et mõlemad pöörduvad koos.
Mõõtkava kopeerija (Copy Scale)	Kopeerib sihtmärgi mõõtkava (valikulise nihkega) valdajale, nii et mõlema mõõtkava teisendatakse koos.
Vahemaa piiraja (Limit Distance)	Piirab valdaja asukoha sihtmärgi suhtes nii, et see püsib alati lähemal/kaugemal/täpsel kaugusel määratud vahemaast.
Asukoha piiraja (Limit Location)	Piirab valdaja asukoha teatud vahemiku piiridesse.
Pöörde piiraja (Limit Rotation)	Piirab valdaja pöörde teatud vahemiku piiridesse.
Mõõtkava piiraja (Limit Scale)	Piirab omaniku mõõtkava teatud vahemiku piiridesse.
Teisendus (Transformation)	Teisendus kasutab mingit sihtmärgi osateisendust (asukoht, pööre või mõõtkava) valdaja sama või erineva osateisenduse juhtimiseks.
Mahuhaldaja (Maintain Volume)	Juhib luu või objekti mahtu.

Järgimise piirajad

Need piirajad püüavad mitmesugustel viisidel juhtida oma valdaja teisendusi nii, et see "osutaks" sihtmärgile või "järgneks" sellele.

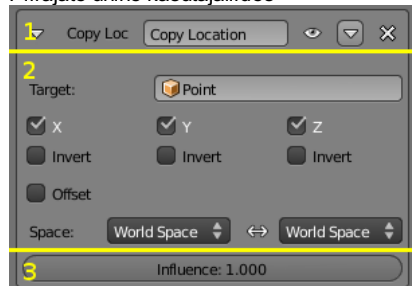
Klammerduspiiraja (<i>Clamp To</i>)	Klammerdab oma valdaja sihtmärgiks antud kõvera külge.
Sumbtrajektor (<i>Damped Track</i>)	
Pöördkinemaatika (<i>Inverse Kinematics</i>)	Ainult luude jaoks. Loob luude ahela, mida sihtmärk pöördkinemaatika abil kontrollib.
Lukustatud trajektooriipiiraja (<i>Locked Track</i>)	Valdaja järgib antud sihtmärki, kuid ühe telje asend on lukustatud.
Splaini pöördkinemaatika (<i>Spline IK</i>)	
Venitaja (<i>Stretch To</i>)	Venitab valdajat antud sihtmärgini.
Järgija (<i>Track To</i>)	Valdaja järgib antud sihtmärki.

Suhteipiirajad

Siia kategooriasse kuuluvad "mitmesugused" piirajad.

Tegevus (<i>Action</i>)	Valdaja käivitab sihtmärgi (draiveri) poolt juhitava tegevuse.
Ülemobjekt (<i>Child Of</i>)	Lubab selektiivselt rakendada alluvussuhte omadusi.
Põrand (<i>Floor</i>)	Kasutab sihtmärgi asukohta (valikuliselt ka pööret), defineerimaks "seina" või "põrandat", millest valdaja ei saa läbi minna.
Raja järgimine (<i>FollowPath</i>)	Valdaja liigub mööda sihtmärkkõverat.
Pöörlemiskese (<i>Pivot</i>)	
Jäiga keha liigend (<i>Rigid Body Joint</i>)	Loob valdaja ning "sihtmärgi" (alluva) vahele jäiga liigendi (hinge...).
Skript (<i>Script</i>)	Kasutab piirajana Pythoni skripti.
Sobitaja (<i>Shrinkwrap</i>)	Piirab valdaja asukoha (muude võimaluste hulgas) sihtmärgi <i>pinnalaotusega</i> .

Piirajate ühine kasutajaliides



Piirajate kasutajaliidese kolm osa

Nagu ka [töõtlejate](#) puhul, võib üks objekt (või luu, vaata [taageldamise peatükki](#)) kasutada korraga mitut piirajat. Seetõttu on piirajad organiseeritud pinusse, mis määrab nende rakendamise järjekorra (ülevalt alla).

Kõigil piirajatel on ühine põhikasutajaliides, mis on paigutatud kolmeks osaks jaotatud alapaneelile:

1. Pääs kõige üldisemate sätetega.
2. Piirajale spetsiifilised sätted.
3. Animatsiooni ja mõju seaded (jäiga keha liigendi (Rigid Body Joint) piirajatel ei ole mõju sätteid).

Piirajate pääs



Piiraja pääs

Alapaneeli pääs on kõigi piirajate puhul sama. Vasakult paremale on siin:

Väike nool

Selle abil saab piiraja sätteid peita/näidata. Näites (*piirajate pinu näidis*) on alluvuspiiraja (Child Of) sellel viisil peidetud.

Piiraja tüüp

See on lihtsalt tekst, mis ütleb, millise piirajaga on parajasti tegu.

Nimelahter

Siin on võimalik piirajale anda vaikimisi määratud nime asemel uus ja arusaadavam.

Sellel seadel on ka teine *tähtis* ülesanne: kui piiraja ei ole toimiv, muutub nimelahter punaseks (nagu on näha pildil "Piiraja pääs"). Kuna enamik piirajaid vajab töötamiseks teist, "sihtmärkobjekti", on need piirajad vahetult pärast loomist nn "punases olekus", kuna Blender ei tea veel, millist objekti või luud sihtmärgina kasutada. See võib juhtuda ka siis, kui sa määrad piirajale vigased sätted: näiteks kui [järgimispiiraja \(Track To\)](#) puhul on vektorid To (suund) ja Up (üles) määratud samale teljele. "Punases olekus" olevaid piirajaid ei võeta pinu rakendamisel arvesse.

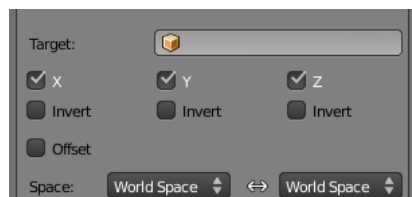
"Üles"/"alla" nupud

Nende abil saab piirajat pinus üles ja allapoole liigutada.

Nupp "X"

Nagu ülalpool näidatud, kustutab see piiraja (eemaldab pinust).

Piirajate sätted



Piirajate alapaneeli keskmises osas on piiraja sätted, sihtmärk ning piiraja koordinaatsüsteem.

Sätete osa on loomulikult igal piiraja tüübil erinev. Kaks asja on enamikel piirajatel siiski samasugused ja neid me käsitleme siin pisut lähemalt.

Sihtmärk

Enamik piirajaid vajavad teist objekti või luud "sihtmärgiks", mis neid juhiks. Sihtmärki saab määrata, valides tema nime lahtris Target (sihtmärk). Kui mõned erandjuhud välja arvata, võib sihtmärgiks olla mistahes tüüpi objekt (kaamera, võre, tühiobjekt...) ja sihtmärkpunktiks saab selle objekti keskpunkt.

Kui objekti (*Target*) lahtrisse trükkida võre või sõrestiku nimi, tekib lahtri alla teine lahter Vertex Group (tipugrupp). Kui jätta see tühjaks, kasutatakse võret või sõrestikku sihtmärgina nagu tavalist objekti. Kui aga sisestada lahtrisse Vertex Group mõni selle võre või sõrestiku tipugrupi nimi, kasutatakse sihtmärgina selle grupi tippude asukohtade keskmist.

Kui objekti (*Target*) lahtrisse trükkida skeleti nimi, tekib samamoodi juurde luu (Bone) lahter. Kui sinna sisestada mõni selle skeleti luu

nimi, kasutab piiraja sihtmärgina selle luu *juurt*. Mõnede piirajate puhul ilmub siis, kui sihtmärgiks on luu, veel pea/saba Head/Tail numbriväli, mis lubab määrata, millises punktis piki luud sihtmärkpunkt asub – alates juurest (**0.0**) kuni tipuni (**1.0**) (tuleta meelde, et praeguses Blenderi kasutajaliideses nimetatakse luude juuri “peadeks” ning tippe “sabadeks”...).

Piiraja koordinaatsüsteem (Space)

Paljude piirajate puhul on võimalik rippmenüüst Owner Space (valdaja koordinaatsüsteem) (paremal) valida, millises koordinaatsüsteemis piiraja mõju arvutatakse ja rakendatakse. Kui see piiraja kasutab ka sihtmärki, saab rippmenüüst Target Space (sihtmärgi koordinaatsüsteem) (vasakul) valida, millises koordinaatsüsteemis sihtmärki arvutatakse. Mõlemal puhul on valikuvõimalused samad, sõltuvalt sellest, kas vastav element (valdaja või sihtmärk) on tavaline objekt või luu:

Local Space (kohalikud koordinaadid)

Luu omadused arvutatakse tema kohalikus koordinaatsüsteemis, st tema puhkeasendi järgi (arvestamata tema ülemate teisendusi luuahelates ja skeletiobjekti teisendusi).

Local With Parent (kohalikud koordinaadid ülemobjektiga) (kehtib ainult luude puhul)

Luu omadused arvutatakse tema kohalikus koordinaatsüsteemis koos võimalike ülemate teisendustega (st ahelas temast eespool asuvate luude teisendustega).

Pose Space (poosi koordinaadid) (kehtib ainult luude puhul)

Luu omadused arvutatakse skelettobjekti kohalikus koordinaatsüsteemis (sõltumatult skeleti enda teisendustest objektirežiimis Object. Seega: kui skeletil pole objekti tasemel teisendusi, käitub poosi koordinaatsüsteem (Pose Space) samamoodi nagu globaalne koordinaatsüsteem (World Space).

Local (Without Parent) Space (kohalikud koordinaadid ilma ülemobjektita) (kehtib ainult objekti puhul)

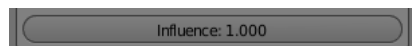
Objekti omadused arvutatakse tema kohalikus koordinaatsüsteemis, *ilma* võimalikust alluvussuhtest tulenevate teisendusteta.

World Space (globaalsed koordinaadid) (vaikesäte)

Objekti või luu omadused arvutatakse globaalses koordinaatsüsteemis. See säte on kõige arusaadavam ja käitub kõige loomulikumalt, kuna ta kasutab alati “visuaalseid” teisendusi (st samasuguseid, nagu sa näed 3D-vaates).

Piiraja koordinaatidest aru saamine ei ole sugugi lihtne (välja arvatud juhul, kui sa oled geomeetriageenius...). Parim lähenemine on erinevate kombinatsioonide katsetamine, näiteks kasutades kaht tühiobjekti (kuna nende teljed on selgelt näha) ja pöörde kopeerimise (Copy Rotation) piirajat (kuna pöörded on erinevate koordinaatsüsteemide visualiseerimiseks kõige näitlikumad teisendused).

Mõju



Mõju

Peaaegu kõigi piirajate all on mõju liugur (Influence), mis reguleerib piiraja mõju oma valdajale. Nagu arvata võib, tähendab väärtus **0.0**, et piirajal ei ole mingit mõju, ja väärtus **1.0** märgib täieliku mõjuga piirajat. Vahepealseid väärtusi kasutades on võimalik panna mitu piirajat koos valdaja sama omadust mõjutama. Pane tähele, et kui piirajal on mingi omaduse üle täielik mõju, siis pinus kõrgemal olevatel sama omaduse piirajatel ei ole mingit mõju.

Kuid parim mõju (*Influence*) liuguri omadus on see, et teda saab animeerida IPO-kõveraga; loe lähemalt [animatsiooni peatüki piirajate osast](#).

Piirajate pinu



Piirajate pinu näidis

Nagu juba öeldud, rakendatakse piirajad piirajate paneelis (Constraints) nähtavas pinus ülevalt alla.

1. Pildil (*Piirajate pinu näidis*) kopeeritakse kõigepealt valdajaobjektile valgusti asukoht.
2. Pöörde kopeerimise piirajat ignoreeritakse (nimi on punane).
3. Järgmine piiraja on alluvuse piiraja (Child Of), mille seaded on hetkel peidetud.
4. Lõpuks on näites kasutatud kuubi suurus piiratud mõõtkava piirajaga (Limit Scale).

Kokkuvõttes on kuubi suurus kõigepealt määratud alluvuspiiraja (Child Of) sihtmärgi poolt talle järgneva mõõtkava piiraja (Limit Scale poolt lubatud vahemikus... Nagu ka töötlejate puhul, on piirajate järjekord väga oluline!

Piirajaid võib pinus üles ja alla liigutada, kasutades väikesed üles/alla nooli nende päises, kohe piiraja nime kõrval. Need nupud on nähtavad ainult vajaduse korral, näiteks kõige ülemisel piirajal on ainult alla suunav nool ning kõige alumisel piirajal ainult üles suunav nool. Kui pinus on ainult üks piiraja, peidetakse mõlemad nupud.

Piiraja lisamine ja eemaldamine

Piiraja lisamiseks võib piirajate paneelis (Constraints) vajutada piiraja lisamise nupule (Add Constraint). Kuvatakse menüü kõigi piirajatega, mida parasjagu aktiivse objekti (või poseerimisrežiimis ka luu) peal on võimalik rakendada. Luu puhul läheb uus piiraja luu piirajate menüüsse. Uus piiraja lisatakse *alati* pinu lõppu.

3D-vaates on võimalik ka:

- Valida tulevane valdaja, seejärel vajutada kiirklahvi **Ctrl+Shift+C**, valida kuvatud menüüst Add Constraint to New Empty Object (lisa piiraja uuele tühiobjektile) ja seejärel soovitud piiraja. Kui valitud piiraja seda nõuab, lisatakse tema sihtmärgiks uus pöörateta tühiobjekt, mis asub valdaja keskpunktis ja millel puudub pööre.
- Kõigepealt tuleb valida tulevane sihtmärk, seejärel tulevane valdaja, vajutada kiirklahvi **Ctrl+Shift+C**, valida kuvatud menüüst Add Constraint to Active Object (lisa aktiivsele objektile piiraja) (või luu puhul Add Constraint to Active Bone (lisa aktiivsele luule piiraja)) ja seejärel soovitud piiraja. Kui valitud piiraja seda nõuab, kasutatakse tema sihtmärgina teisena valitud objekti või luud.

Pane tähele, et need menüüd ei paku alati kõiki võimalikke piirajaid...

Piiraja eemaldamiseks tuleb piirajate paneelis (Constraints) vajutada tema päises asuvat nuppu "X". Sa saad valitud objekti(de)lt kõik piirajad eemaldada, valides menüüst Object » Constraints » Clear Constraints (piirajad -> kustuta piirajad) (või Pose » Constraints » Clear Constraints (poos -> piirajad -> kustuta piirajad)) või kiirklahviga **Ctrl+Alt+C**.

Asukoha kopeerimise piiraja (*Copy Location*)

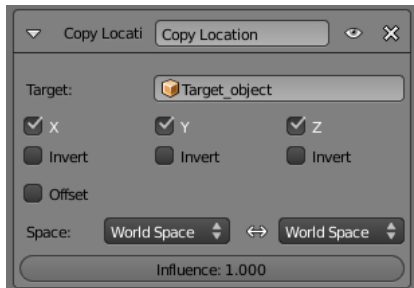
Kirjeldus

Asukoha kopeerimise piiraja (*Copy Location*) sunnib oma valdajat paiknema sihtmärgiga samas kohas.



Pane tähele, et *seotud* luu jaoks ei oma see piiraja mingit efekti, kuna tema valdaja ehk siis vastava luu juure asukoha määrab ülema tipp.

Valikud



Asukoha kopeerija paneel *Copy Location*

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgi lahtris Target on sihtmärgiks valitud skelett (*Armature*), kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne luu.

Head/Tail (pea/saba)

Kui sihtmärgiks on määratud luu, kuvatakse uus lahter, kus on võimalik määrata sihtmärgi täpne asukoht luu peal.

Vertex Group (tipugrupp)

Kui sihtmärgiks on määratud võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp.

X, Y, Z

Need nupud määravad, milliseid telgi (st koordinaate) piiratakse. Vaikesättena mõjub piiraja kõigile telgedele.

Invert (pööra ümber)

Nupud Invert muudavad neile vastavate koordinaatide väärtused vastupidisteks.

Offset (nihe)

Kui see on sisse lülitatud, võimaldab ta omanikku sihtmärgi asukoha suhtes nihutada (kasutades tema hetkel kehtivat teisendust).

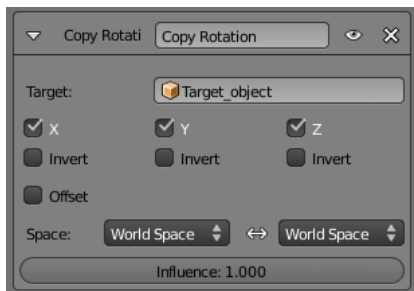
Space (koordinaatsüsteem)

Selle piiraja puhul on võimalik valida, millises koordinaadistikus tema valdaja ning sihtmärgi teisendusomadusi arvutatakse.

Pöörde kopeerimise piiraja (*Copy Rotation*)

Pöörde kopeerimise piiraja (Copy Rotation) paneb oma valdaja ühilduma oma pöörde sihtmärgi pöördega.

Valikud



Pöörde kopeerimise paneel Copy Rotation

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgi lahtris Target on sihtmärgiks valitud skelett (Armature), kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne luu.

Head/Tail (pea/saba)

Kui sihtmärgiks on määratud luu, kuvatakse uus lahter, kus on võimalik määrata sihtmärgi täpne asukoht luu peal.

Vertex Group (tipugrupp)

Kui sihtmärgiks on määratud võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp.

X, Y, Z

Need nupud määravad, milliseid telgi piiratakse. Vaikesättena mõjub piiraja kõigile telgedele.

Invert (pööra ümber)

Invert nupud muudavad vastavate telged pöörde väärtused vastupidisteks.

Offset (nihe)

Kui see on sisse lülitatud, võimaldab ta valdaja pööret sihtmärgi suhtes muuta (kasutades tema hetkel kehtivat teisendust).

Space (koordinaatsüsteem)

Selle piiraja puhul on võimalik valida, millises koordinaadistikus tema valdaja ning sihtmärgi teisendusomadusi arvutatakse.

Mõõtkava kopeerimise piiraja (*Copy Scale*)

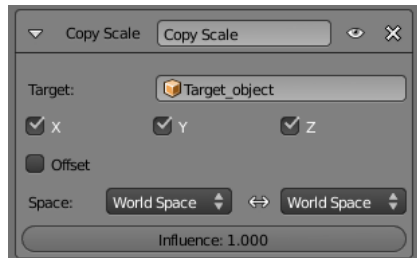
Kirjeldus

Mõõtkava kopeerimise piiraja Copy Scale sunnib oma valdajat ühildama oma mõõtkava sihtmärgi omaga.



Me räägime siin **mõõtkavast**, mitte **suurusest**! Sul võib vabalt olla kaks objekti, üks suurem kui teine, aga neil mõlemal on ühesugune mõõtkava. See kehtib poseerimisrežiimis Pose ka luude puhul: neil kõigil on puhkeasendis sama mõõtkava hoolimata sellest, milline on nende nähtav pikkus.

Valikud



Mõõtkava kopeerimise paneel Copy Scale

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgi lahtris Target on sihtmärgiks valitud skelett (Armature), kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne luu.

Head/Tail (pea/saba)

Kui sihtmärgiks on määratud luu, kuvatakse uus lahter, kus on võimalik määrata sihtmärgi täpne asukoht luu peal.

Vertex Group (tipugrupp)

Kui sihtmärgiks on määratud võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp.

X, Y, Z

Need nupud määravad, milliseid telgi piiratakse. Vaikesättena mõjub piiraja kõigile telgedele.

Offset (nihe)

Kui see on sisse lülitatud, võimaldab ta omaniku mõõtkava sihtmärgi suhtes muuta (kasutades tema hetkel kehtivat teisendust).

Space (koordinaatsüsteem)

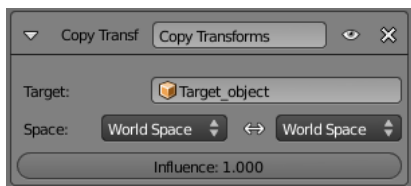
Selle piiraja puhul on võimalik valida, millises koordinaadistikus tema valdaja ning sihtmärgi teisendusomadusi arvutatakse.

Teisenduste kopeerimise piiraja (*Copy Transforms*)

Kirjeldus

Teisenduste kopeerimise piiraja (Copy Transforms) paneb oma valdaja ühildama oma teisenduse sihtmärgi omaga.

Valikud



Teisenduste kopeerimise paneel Copy Transforms

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgi lahtris Target on sihtmärgiks valitud skelett (Armature), kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne luu.

Head/Tail (pea/saba)

Kui sihtmärgiks on määratud luu, kuvatakse uus lahter, kus on võimalik määrata sihtmärgi täpne asukoht luu peal.

Vertex Group (tipugrupp)

Kui sihtmärgiks on määratud võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp.

Space (koordinaatsüsteem)

Selle piiraja puhul on võimalik valida, millises koordinaadistikus tema valdaja ning sihtmärgi teisendusomadusi arvutatakse.

Kauguse piiraja (*Limit Distance*)

Kirjeldus

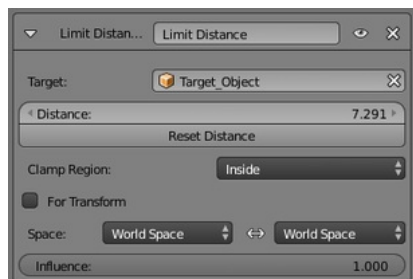
Kauguse piiraja (Limit Distance constraint) sunnib oma valdajat püsima sihtmärgi suhtes etteantud vahemaast kaugemal, lähemal või täpselt sellel kaugusel. Teisisõnu piiratakse valdaja asukoht sihtmärgi keskpunkti ümber asetsevast kerast väljaspoole, sissepoole või selle pinnale.

Kui sa määrad (uue) sihtmärgi, pannakse kauguse (Distance) väärtuseks automaatselt valdaja ja sihtmärgi vahemaa.



Pane tähele, et kui sa kasutad seda piirajat *seotud* luul, ei ole tal mingit mõju, sest valdaja juure asukoha määrab tema ülema tipp.

Valikud



Kauguse piiraja paneel Limit Distance

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgi lahtris Target on sihtmärgiks valitud skelett (Armature), kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne luu.

Head/Tail (pea/saba)

Kui sihtmärgiks on määratud luu, kuvatakse uus lahter, kus on võimalik määrata sihtmärgi täpne asukoht luu peal.

Vertex Group (tipugrupp)

Kui sihtmärgiks on määratud võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp.

Distance (kaugus)

See numbrilahter määrab piirkauguse ehk teisisõnu piirava kera raadiuse.

Reset Distance (lähtesta kaugus)

Kui sellele väikesele nupule vajutada, algväärtustatakse kaugus Distance nii, et ta vastab valdaja ja sihtmärgi tegelikule vahemaale (st nendevahelisele kaugusele enne piiraja rakendamist).

Clamp Region (klammerdusala)

Piiramisrežiimi (Limit Mode) rippmenüü võimaldab valida, kuidas piiraja tõlgendab vahemaa (Distance) väärtust ja sihtmärgi keskpunkti:

Inside (sees) (vaikesäte)

Valdaja piiratakse kera *sisse*.

Outside (väljas)

Valdaja piiratakse kerast *väljaspoole*.

Surface (pinnal)

Valdaja piiratakse kera *pinnale*.

Asukoha piiraja (*Limit Location*)

Kirjeldus

Objekti või *sidumata* luud võib stseenis liigutada piki X-, Y- ja Z-telgi. See piiraja määrab igal teljel võimaliku asukoha minimaalse ja maksimaalse väärtuse.

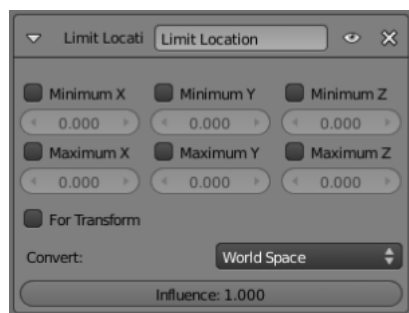
Objekti asukoha piirid arvutatakse tema keskpunktist ja luu puhul tema juurest.

Tasub märkida, et ehkki valdaja visuaalne ja renderdatud asukoht on piiratud, lubab valdaja andmeblokk siiski (vaikimisi) objekti või luu asukoha koordinaatide väärtusi väljaspool sätestatud maksimumi ja miinimumi. Seda võib näha teisenduste paneelis (Transform Properties) (N). Kui haarata valdajast hiirega kinni ja üritada liigutada teda väljapoole lubatud piirkonda, jääb ta 3D-vaates ning renderdustes piiraja poolt määratud vahemikku, kuid sisemiselt võivad tema koordinaatide väärtused olla väljaspool seda. Piiraja eemaldamisel hüppab selle endine omanik oma sisemiselt määratud asukohta.

Ja samamoodi: kui selle piiraja valdaja sisemine asukoht on väljaspool lubatud vahemikku, ei näi tema lubatud alasse tagasi nihutamine mitte midagi tegevat, enne kui tema sisemised koordinaadid jõuavad vahemiku piiridesse (välja arvatud juhul, kui valik For Transform (teisenduse jaoks) on sisse lülitatud; vaata täpsemalt altpoolt).

Telje minimaalse asukoha võrdsustamine maksimaalse asukohaga lukustab valdaja liikumise piki seda telge... Kuigi nii saab teha, on teisendusomaduste (Transformation Properties) telje lukustamine tõenäoliselt lihtsam!

Valikud



Asukoha piiraja paneel Limit Location

Minimum X, Minimum Y, Minimum Z (minimaalne X, minimaalne Y, minimaalne Z)

Need nupud lülitavad sisse valdaja keskpunkti miinimumväärtuse valitud koordinaatsüsteemi X-, Y- ja Z-teljel.

Numbriahet nende all määrab nende piirangute väärtuse.

Kui mõõtkava minimaalne väärtus on suurem maksimaalsest väärtusest, käitub piiraja nii, nagu oleks minimaalne võrdne maksimaalsega.

Maximum X, Maximum Y, Maximum Z (maksimaalne X, maksimaalne Y, maksimaalne Z)

Need nupud lülitavad sisse valdaja keskpunkti maksimumväärtuse valitud koordinaatsüsteemi X-, Y- ja Z-teljel.

Samad valikud, kui ülalpool mainitud.

For Transform (teisenduse jaoks)

Me nägime, et vaikimisi võivad valdaja koordinaadid asuda väljaspool piire (nagu on näha teisenduste paneelis), ehkki valdaja liikumine on visuaalselt piiratud. Kui sa aga lülitad selle nupu sisse, ei ole see enam võimalik: valdaja teisendus on sel juhul samuti piiratud.

Pane tähele, et ainuüksi piiraja rakendamine ei muuda otseselt valdaja koordinaate: et koordinaatide väärtused lõigataks etteantud vahemikku, pead sa valdajast ühel või teisel viisil kinni haarama.

Convert (muuda)

See piiraja võimaldab sul valida, millises koordinaatsüsteemis tema valdaja teisendusi arvutatakse.

Pöörde piiraja (*Limit Rotation*)

Kirjeldus

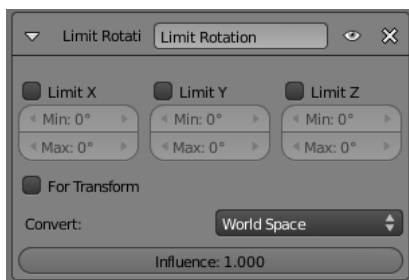
Objekti või luud saab pöörata ümber tema X-, Y- või Z-telje. See piiraja määrab minimaalse ja maksimaalse lubatud pöördenurga iga telje ümber.

Tasub märkida, et kuigi valdaja visuaalne ja renderdatud pööre on piiratud, lubab valdaja andmeblokk siiski (vaikimisi) objekti või luu pöördenurga väärtusi väljaspool sätestatud maksimumi ja miinimumi. Seda võib näha teisendusomaduste paneelis (Transform Properties) kiirvalikuklahviga (N). Kui valdajat proovitakse pöörata rohkem kui piiraja poolt lubatud, sunnitakse ta visuaalselt ning renderdustes piiraja poolt lubatud maksimaalse pöörde väärtuseni, ent sisemiselt võivad tema pöördenurgad olla nendest suuremad. Piiraja eemaldamisel hüppab selle endine omanik oma sisemiselt määratud pöörde väärtuseni.

Kui valdaja sisemine pööre on määratud piiraja lubatust kaugemale, ei ole tema tagasipööramisest näiliselt mingit kasu, enne kui pööre jõuab tagasi piiraja poolt lubatud alasse (kui pole aktiveeritud teisenduse valik For Transform (teisenduse jaoks), mida kirjeldatakse täpsemalt allpool).

Telje minimaalse pöörde võrdsustamine maksimaalse pöördega lukustab valdaja pööremise selle telje ümber... Kuigi nii saab teha, on teisendusomaduste (Transformation Properties) telje lukustamine tõenäoliselt lihtsam!

Valikud



Pöörde piiraja paneel Limit Rotation

Limit X, LimitY, LimitZ (X, Y, Z piir)

Need nupud aktiveerivad pöörde piiraja vastavalt valdaja X-, Y- ja Z-telje suhtes ja valitud koordinaadistikus.

Min ja Max numbriväljad neist paremal pool määravad minimaalsed ja maksimaalsed pöördenurga väärtused.

Panelele, et:

- Kui pöörde minimaalne väärtus on suurem maksimaalsest väärtusest, käitub piiraja nii, nagu oleks minimaalne võrdne maksimaalsega.
- Erinevalt [asukoha piirajast](#) ei ole pöördel võimalik piirangute ülemist ja alumist väärtust eraldi aktiveerida.

For Transform (teisenduse jaoks)

Me nägime, et vaikimisi võivad valdaja pöördenurgad asuda väljaspool piire (nagu on näha teisenduste paneelis), ehkki valdaja pööremine on visuaalselt piiratud. Kui sa aga lülitad selle nupu sisse, ei ole see enam võimalik: valdaja teisendus on sel juhul samuti piiratud.

Panelele, et ainuüksi piiraja rakendamine ei muuda otseselt pöörde väärtust, piirangute kehtima hakkamiseks tuleb valdajat mingis suunas pöörata...

Convert (muuda)

See piiraja võimaldab sul valida, millises koordinaatsüsteemis tema valdaja teisendusi arvutatakse.

Mõõtkava piiraja (*Limit Scale*)

Kirjeldus

Objekti või luu mõõtkava saab teisendada X-, Y- ja Z-telje suunas. See piiraja määrab minimaalse ja maksimaalse lubatud mõõtkava iga telje suhtes.



See piiraja ei luba negatiivseid väärtusi (kuigi neid võiks kasutada näiteks objekti peegeldamiseks) – kui sa lisad objektile või luule mõõtkava piiraja, siis isegi kui piirang pole ühegi telje suhtes sisse lülitatud ega ka For Transform nupp aktiveeritud, muudetakse kõik negatiivsed mõõtkava väärtused automaatselt positiivseteks. Ja piiride väärtusteks saab sisestada ainult rangelt positiivseid arve.

Tasub märkida, et ehkki valdaja visuaalne ja renderdatud mõõtkava on piiratud, lubab valdaja andmeblokk siiski (vaikimisi) objekti või luu mõõtkava väärtusi väljaspool sätestatud maksimumi ja miinimumi (niikaua, kui need väärtused on positiivsed). Seda võib näha teisendusomaduste paneelis (Transform Properties) (N). Kui valdaja mõõtkava proovitakse teisendada rohkem kui piiraja poolt lubatud, sunnitakse ta visuaalselt ning renderdustes piiraja poolt lubatud maksimaalse mõõtkava piiridesse, ent sisemiselt võivad tema mõõtkava väärtused olla nendest suuremad. Piiraja eemaldamisel hüppab selle endine omanik oma sisemiselt määratud mõõtkava väärtuseni.

Ja samamoodi: kui valdaja sisemine mõõtkava on määratud piiraja poolt lubatud suuremaks/väiksemaks, ei ole mõõtkava muutmise näiliselt mingit kasu, enne kui mõõtkava jõuab tagasi piiraja poolt lubatud alasse (kui pole aktiveeritud teisenduse valik For Transform, mida kirjeldatakse täpsemalt allpool, või kui mõned valdaja mõõtkava väärtused pole negatiivsed).

Telje minimaalse mõõtkava võrdsustamine maksimaalse mõõtkavaga lukustab valdaja mõõtkava teisendamise selle telje suhtes. Kuigi nii saab teha, on teisendusomaduste (Transformation Properties) telje lukustamine tõenäoliselt lihtsam!

Valikud



Mõõtkava piiraja paneel Limit Scale

Minimum/Maximum X, Y, Z (miinimum/maksimum X, Y, Z)

Need nupud aktiveerivad mõõtkava piiraja miinimumi ja maksimumi valitud koordinaatsüsteemi X-, Y- ja Z-telgedel.

Min ja Max numbriväljad neist paremal pool määravad minimaalsed ja maksimaalsed pöördenurga väärtused.

Kui mõõtkava minimaalne väärtus on suurem maksimaalsest väärtusest, käitub piiraja nii, nagu oleks minimaalne võrdne maksimaalsega.

For Transform (teisenduse jaoks)

Me nägime, et vaikimisi võivad valdaja sisemise mõõtkava väärtused asuda väljaspool piire (nagu on näha teisenduste paneelis) tingimusel, et nad on positiivsed, ehkki valdaja mõõtkava muutmine on visuaalselt piiratud. Kui sa aga lülitad selle nupu sisse, ei ole see enam võimalik, valdaja teisendus on sel juhul samuti piiratud.

Pane tähele, et piiraja kehtestamine ei muuda otseselt mõõtkava väärtust, piirangute kehtima hakkamiseks tuleb valdaja mõõtkava ka mingis suunas teisendada.

Convert (muuda)

See piiraja võimaldab sul valida, millises koordinaatsüsteemis tema valdaja teisendusi arvutatakse.

Mahu haldamise piiraja (*Maintain Volume*)

Kirjeldus

Mahu haldamise piiraja (*Maintain Volume*) piirab võre või luu mahu suurst võrreldes tema algse mahuga.

Valik



Mahu haldamise paneel *Maintain Volume*

Free X/Y/Z (vaba X/Y/Z)

Telg, mööda mida saab objekti mõõtkava vabalt teisendada.

Volume (maht)

Luu maht puhkeasendis. Vaikesättena on see 1.0.

Space (koordinaatsüsteem)

Selle piiraja võimaldab valida, millises koordinaadistikus tema valdaja ning sihtmärgi teisendusi arvutatakse.

Loe lisaks

- [Harkymani kirjeldust mahuhaldaja arendusest](#), 2010 märts.

Teisenduse piiraja (*Transformation*)

Kirjeldus

See piiraja on keerulisem ning paindlikum kui teised "teisenduste" piirajad. Teisenduse piiraja võimaldab sul seostada ühe valdaja alamteisenduse tüübi (st asukoha, pöörde või mõõtkava) sihtmärgi sama või teist tüüpi alamteisendusega, määraates väärtuste vahemiku (mis võib valdajal ning sihtmärgil olla erinev). Sa saad piiranguid ka telgede vahel vahetada ning etteantud vahemike väärtused ei ole enam otseselt piirid, vaid pigem "tähised", mis määravad sisendi (sihtmärgi) ja väljundi (valdaja) väärtuste seose.

Seega võid sa näiteks kasutada sihtmärgi asukohta X-teljel, juhtimaks valdaja pööret ümber Z-telje, defineerides, et **1 BU** piki sihtmärgi X-telge vastab **10°** pöördele ümber valdaja Z-telje. Seda kasutatakse sageli hammasrataste (vaata märkus allpool) jaoks ja asukohast sõltuvate pöörete määramiseks.

Hammasrattad

Kahjuks ei tööta see piiraja hammasratastega, sest sisemiste arvutusprobleemide tõttu ei anna ta ilusaid tulemusi.

Valikud



Teisenduse piiraja paneel Transformation

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgi lahtis Target on sihtmärgiks valitud skelett (Armature), kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne luu.

Head/Tail (pea/saba)

Kui sihtmärgiks on määratud luu, kuvatakse uus lahter, kus on võimalik määrata sihtmärgi täpne asukoht luu peal.

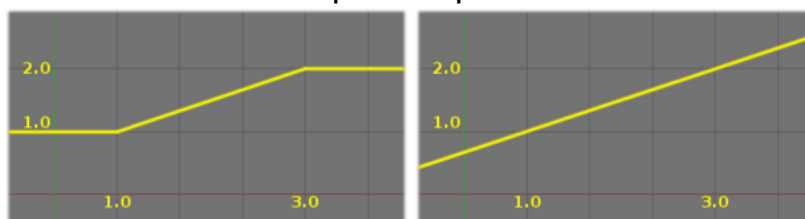
Vertex Group (tipugrupp)

Kui sihtmärgiks on määratud võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp.

Extrapolate (ekstrapoleerimine)

Vaikesättena piiravad min ja max väärtused sisendi ja väljundi võimalike väärtuste ulatust ning kõik neist suuremad või väiksemad väärtused "lõigatakse" täpselt piiridele vastavaks. Selle nupu aktiveerimisel pole min ja max väärtused enam ranged piirangud, vaid pigem "märgised", mis määravad proportsionaalse (lineaarse) seose sisendi ja väljundi vahel. Illustreerime seda kahe graafikuga (Ekstrapoleerimise põhimõte). Nagu näha piltidelt, on sisendi (x-teljel) vahemikuks määratud [1.0, 4.0] ja sellele vastavaks väljundi vahemikuks (y-teljel) [1.0, 2.0]. Kollane kõver esindab sisendi ja väljundi vahelist seost.

Ekstrapoleerimise põhimõte.



Extrapolate (Ekstrapoleerimine) välja lülitatud: väljundi väärtused on piiratud vahemikku [1.0, 2.0].

Extrapolate (Ekstrapoleerimine) sisse lülitatud: väljundi väärtused järgivad proportsionaalselt sisendi väärtusi.



Pane tähele, et:

- Seostades alamteisendust asukohaga (st nupp Loc Destination on aktiivne), lisatakse piiraja arvutamisel tulemusele selle valdaja praegune asukoht (samamoodi nagu nupu Offset (nihe) aktiveerimisel [Copy Location \(Asukoha kopeerija\) puhul](#)).
- Samas aga, seostades alamteisendust pöörde või mõõtkavaga, kirjutatakse piiraja arvutamisel tema valdaja olemasolev pööre või mõõtkava üle.

- Kasutades sisendina sihtmärgi pööret, "paigutab" piiraja selle alati vahemikku $[-180^\circ, 180^\circ]$, hoolimata tema tegelikust väärtustest (näiteks kui sihtmärgi pööre on **420°** ümber oma X-telje, kasutab piiraja X-sisendiks väärtust $((420 + 180) \bmod 360) - 180 = 60^\circ \dots$). Sellepärast ei töötagi see piiraja hammasratastega!
- Kasutades sisendina sihtmärgi mõõtkava, kasutab piiraja väljundi arvutamisel selle absoluutväärtust, vaatamata tegelikule väärtusele (st muudab negatiivsed väärtused positiivseks).
- Kui välja min väärtus on suurem kui välja max väärtus, loetakse mõlemad võrdseks max väärtusega. See tähendab, et "pööratud" seoste loomine ei ole võimalik...

Source (sisend)

Sihtmärgi sisendi seaded.

Kolm lüliti – Loc (asukoht), Rot (pööre) ja Scale (mõõtkava) – on vastastikku üksteist välistavad ja määravad ära, millist allteisendust sisendina kasutatakse.

X-, Y- ja Z: min ning max numbriväljad määravad sõltumatult iga telje jaoks minimaalse ja maksimaalse sisendväärtuse vahemiku. Kui minimaalne väärtus on suurem maksimaalsest väärtusest, käitub piiraja nii, nagu minimaalne oleks võrdne maksimaalsega.

Destination (väljund)

Valdaja väljundi seaded.

- Kolm lüliti – Loc (asukoht), Rot (pööre) ja Scale (mõõtkava) – on vastastikku üksteist välistavad ja määravad ära, millisele allteisendusele väljund rakendub.
- Kolm telgede seose rippmenüüd (Axis Mapping) võimaldavad valida, milline sisendi telg tuleb seostada vastavalt (ülevallt alla) väljundi (valdaja) X-, Y- ja Z-telgedega.
- min ja max numbriväljad määravad sõltumatult iga telje jaoks minimaalse ja maksimaalse väljundväärtuse vahemiku.. Kui minimaalne väärtus on suurem maksimaalsest väärtusest, käitub piiraja nii, nagu minimaalne oleks võrdne maksimaalsega.

Space (koordinaatsüsteem)

Selle piiraja puhul on võimalik valida, millises koordinaadistikus tema valdaja ning sihtmärgi teisendusomadusi arvutatakse.

Klammerduspiiraja (*Clamp To*)

Kirjeldus

Klammerduspiiraja (*Clamp To*) on eriti kasulik asjade liigutamiseks mööda keerulisi ja pikki trajektoore, mida oleks käsitsi võtmekaadrede määramine raske saavutada.

Klammerduspiiraja (*Clamp To*) tööpõhimõte on väga samane [trajektooriirajale \(Follow Path\)](#) ühe olulise erinevusega: kui trajektooriiraja kasutab sihtmärkkõvera IPO aega, siis klammerduspiiraja loeb oma valdaja tegelikud asukohaparametrid (need, mida näeb paneelil Transform Properties (teisenduse omadused), kiirklahv N) ja otsustab selle, kuhu valdaja paigutada, "projekteerides" tema asukoha sihtmärkkõverale.

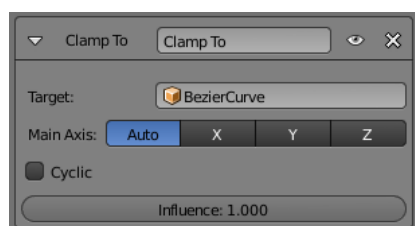
Nagu enamikel juhtudel, on siin nii omad head kui ka vead.

Positiivne külg on see, et klammerduspiirajaga töötades on kergem jälgida valdaja tegevust: töö käib 3D-vaates, mis on palju täpsem kui IPO-kõveratel kaadrede ümber tõstes ning animatsiooni pidevalt üle korrates.

Negatiivne külg on see, et erinevalt [trajektooriirajast](#) pole klammerduspiiraja kasutamisel võimalik järgida valdaja pööret, et saavutada kurvides kallutamist. Et seda alati aga vaja pole, on klammerduspiirajat enamasti mugav kasutada, lisades vajalikud pöörded mõnel teisel viisil.

Kokkuvõttes võib öelda, et klammerduspiirajaga on kõverat järgivat liikumist palju lihtsam animeerida kui trajektooriirajaga; ja isegi kui ei ole lihtsam, on ta arvessevõetav alternatiiv – loomulikult sõltuvalt isiklikest eelistustest.

Valikud



Klammerduspiiraja paneel Clamp To

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleleta ei tööta (on punases olekus).

Main Axis (peatelg)

See nupurühm määrab, millise globaalse telje (X, Y või Z) järgi valdaja sihtmärkkõverale paigutatakse.

Ükski nendest valikutest pole vale, nii et vali telg, millega on kõige lihtsam töötada või mis hetkeolukorras kõige paremini töötab. Hea mõte on valida see telg, mida pidi mõttes sihtmärkkõver on kõige pikem, nii et objekti globaalsed ja piiratud asukohad on üksteisele võimalikult sarnased. Võib ka kasutada valikut Auto (automaatne) ja lasta Blenderil endal parim telg valida.

Cyclic (kordus)

Vaikesättena jääb valdaja, kui ta on jõudnud sihtmärkkõvera lõppu, sinna pidama, kui sa teda samas suunas edasi nihutad. Kui valik Cyclic (kordus) on aktiivne, hüppab valdaja kõvera lõppu jõudes koheselt tagasi selle algusesse.

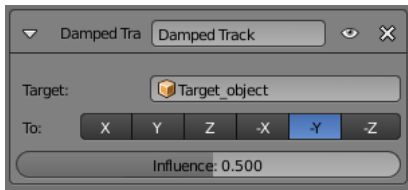
See on mõistagi mõeldud peamiselt suletud kõverate jaoks (ringid jms), kuna ta võimaldab valdajal sama kõverat korduvalt läbida.

Sumbtrajektoori piiraja (*Damped Track*)

Kirjeldus

Sumbtrajektoori piiraja (Damped Track) orienteerib objekti ühe lokaalse koordinaattelje alati sihtmärgi suunas.

Valikud



Sumbtrajektoori paneel Damped Track

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgi lahtris Target on sihtmärgiks valitud skelett (Armature), kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne luu.

Vertex Group (tipugrupp)

Kui sihtmärgiks on määratud võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp.

To (suund)

Sihtmärgi järgiv telg (vaikesättena Y), st valdaja lokaalne koordinaattelg, mis on alati suunatud sihtmärgi poole. Negatiivsed väärtused panevad järgiva telje osutama sihtmärgist eemale.

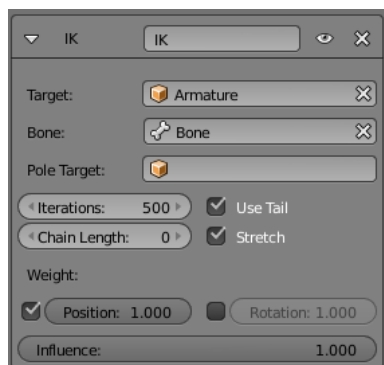
Pöördkinemaatika piiraja (*Inverse Kinematics*)

Kirjeldus

Pöördkinemaatika piiraja (Inverse Kinematics) teostab "pöördkinemaatilist" skeleti poseerimistehnikat. Seetõttu saab seda kasutada ainult luude puhul.

Sellest piirajast on pikemalt juttu taageldamise peatüki [pöördkinemaatika osas](#).

Valikud



Pöördkinemaatika paneel Inverse Kinematics

Target (sihtmärk)

Peab olema skelett

Bone (luu)

Skeletti kuuluv luu

Pole Target (poolussihtmärk)

Keskmise pöörde suunda määrav objekt

Iterations (iteratsioonid)

Maksimaalne iteratsioonide arv IK lahendamiseks

Chain Length (ahela pikkus)

IK ahelasse kuuluvate luude arv Kõikide luude kaasamiseks määra selle väärtuseks 0.

Use Tail (kasuta saba)

Paiguta luu saba viimase elemendina ahelasse.

Stretch (venitus)

Luba IK venitusi

Weight (kaal)

Position (asukohat)

IK puu jaoks: sihtmärgi asukoha kaal.

Rotation (pööre)

Ahel järgib sihtmärgi pööordeid

Target (sihtmärk)

Sihtmärgita IK jaoks lülita see välja

Rotation (pööre)

Ahel järgib sihtmärgi pööordeid

Lukustatud trajektoori piiraja (*Locked Track*)

Kirjeldus

Lukustatud trajektoori piiraja (Locked Track) omadusi on keeruline seletada nii pildis kui ka tekstis. Põhimõtteliselt on tegu [järgimise piirajaga \(Track To\)](#), mille üks telg on lukustatud, st seda ei saa pöörata (tema suunda muuta). Seetõttu saab valdaja oma sihtmärki jälitada ainult ümber selle telje pööreldes ja – välja arvatud juhul, kui sihtmärk asub valdaja keskpunkti läbival selle teljega ristuv tasandil – ei saa üldjuhul täpselt sihtmärgi suunas osutada.

Võtame kõige sobivama näite päris maailmast: kompassi. Kompass võib end pöörata oma sihtmärgi üldises suunas (magnetiline põhjapoolus või lähedal asuv magnet), ent ta ei saa täpselt oma sihtmärgi suunas osutada, kuna ta pöörleb nagu ratas telje otsas. Kui kompass on asetatud lauale ning magnet asub otse tema kohal, pole kompassil kuidagi võimalik tema suunas osutada. Kui me liigutame magnetit rohkem kompassi ühe külje poole, pole *otsene* osutamine ikkagi võimalik, kuid ta saab näidata üldist suunda, jäädes ikkagi pöörlemisteljega piiratuks.

Kasutades lukustatud trajektoori piirajat (Locked Track), võib sihtmärki kujutada ette magnetina ning valdajat kompassina. Lukustatud telg (Lock) käitub teljena, mille ümber valdaja pöörleb, ning suunatelg (To) käitub kompassinõelana. Milline telg kuidas käitub, on sinu otsustada!

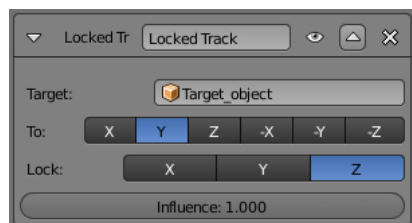
Kui sa ei saa aru, mida selle piiraja nupud teevad, loe vihjeid, need selgitavad selle toimimist päris hästi. Kui sa ei tea, mis suunas sinu objekti teljed on orienteeritud, lülita sisse nupp Axis (telg) menüü Object (objekt) paneelil Draw (joonista). Või kui sa töötad luudega, lülita sisse nupp Axes (teljed) menüü Armature (skelett) paneelil Display (esitus).

See piiraja on loodud koostööks järgimispiirajaga (Track To). Kui nende kahe piiraja teljed õigesti seadistada, võib järgimispiirajat (Track To) kasutada telje suunamiseks peamisele sihtmärgile ning lukustatud trajektoori piirajat (Locked Track) valdaja pööramiseks ümber oma telje sekundaarse sihtmärgi suunas.

Need piirajad töötavad hästi ka 2D-tahvite tegemisel (*billboarding*).

Neid teemasid käsitletakse pikemalt [2.49 BSoD järgimise õppetükis](#).

Valikud



Trajektoori piiraja paneel Locked track

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma sellela ei tööta (on punases olekus).

To (suund)

Sihtmärki järgiv telg (vaikesättena Y), st valdaja lokaalne koordinaattelg, mis on alati suunatud sihtmärgi poole. Negatiivsed väärtused panevad järgiva telje osutama sihtmärgist eemale.

Lock (lukustatud telg)

Valdaja telg, mille suunda sihtmärgi järgimisel ei muudeta (vaikimisi Z).



Kui sa valid järgivaks (To) ja lukustatud (Lock) teljeks sama telje, siis piiraja ei toimi ning on punases olekus.

Splaini IK piiraja (*Spline IK*)

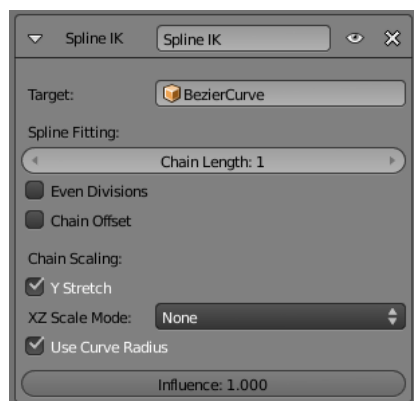
Kirjeldus

Splaini IK piiraja Spline IK joondab luude ahela mööda kõverat. Spline IK on hindamatu tööriist taageldajale, sest ta võimaldab ühendada kõverate abil saavutatavad kaunid ja sujuvad kujundid luude poolt pakutava täpse juhtimisega. Ta on eriti sobiv paindlike kehaosade, näiteks sabade, tundlate ja selgroogude, ning elutute esemete, näiteks kõite taageldamiseks.

Spline IK seadistamiseks on vaja omavahel ühendatud luude ahelat ning nende piiramiseks kasutatavat kõverat.

1. Olles selekteerinud ahela viimase luu, lisa splaini IK piiraja (Spline IK) luude piirajate (Bone Constraints) kaardilt omaduste redaktoris (Properties Editor).
2. Määra ahela pikkuseks (Chain Length) luude arv ahelas (alates valitud luust, see kaasa arvatud), mida see piiraja peaks mõjutama.
3. Lõpuks tuleb sihtmärgiks (Target) määrata kõver, mis hakkab ahelat juhtima.

Valikud



Splaini IK paneel Spline IK

Target (sihtmärk)

Sihtmärgiks valitud kõver

Spline Fitting (splaini sobitamine)

Chain Length (ahela pikkus)

Kui mitu luud on ahelas

Even Division (ühtlane jaotus)

Ignoreerib luude suhtelist pikkust nende kõverale sobitamisel

Chain Offset (ahela nihe)

Nihutab tervet ahelat selle juure suhtes

Chain Scaling (ahela mõõtkava teisendamine)

Y stretch (Y venituse)

Venitab luude Y-telge kõveraga sobitamiseks

XZ Scale Mode (XZ mõõtkava teisendamine)

None (puudub)

Vaikesäte: X- ja Y-telje mõõtkava ei teisendata

Bone Original (luu algne)

Kasutab luude algset mõõtkava

Volume Preservation (mahu säilitamine)

Teisendab X- ja Z-telgede mõõtkava Y mõõtkava pöördväärtusega

Use Curve Radius (kasuta kõvera raadiust)

Lisaks X ja Z mõõtkava teisendamise režiimile muudetakse vastavat (X ja Z) mõõtkava kõvera algus- ja lõpp-punkti keskmise raadiuse järgi.

Loe lisaks

- [Blender.org 2.56 väljalaskelogi splaini IK kohta](http://blender.org/2.56/väljalaskelogi/splaini-ik-kohta)

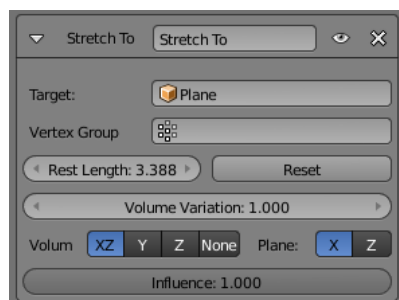
Venitamispiiraja (*Stretch To*)

Venitamispiiraja (*Stretch To*) sunnib oma valdajat pöörama ning mõõtkava teisendama oma Y-telje sihtmärgini. Suuna järgimine toimib samal põhimõttel kui [järgimise piirajal \(Track To\)](#). Lisaks eeldab venitamispiiraja, et järgimis- ja venitusteljeks on Y-telg, ega lase seda muuta.

Tal on lisaks mõned mahu varieerimise võimalused, nii et valdajat surutakse kokku, kui sihtmärk lähemale liigub, ning venitatakse õhemaks, kui sihtmärk kaugemale liigub. Pane siiski tähele, et selle läbi ei säilitata mitte valdaja tegelikku, vaid mõõtkava teisendamise väärtuste kaudu defineeritud virtuaalset mahtu. Seetõttu töötab see piiraja ka mittemahuliste objektidega, näiteks tühiobjektid, 2D-võred, pinnad ja kõverad.

Luude puhul teisendab mõõtkava "mahuvariatsioon" neid piki kohalikke telgi (luu kohalik Y-telg on temaga alati samasihiline, kulgedes juurest tipuni).

Valikud



Venitamispiiraja paneel Stretch To

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgiks (Target) on skelett, kuvatakse uus lahter, kus on võimalik sihtmärgiks valida konkreetne luu.

Head/Tail (pea/saba)

Kui sihtmärgiks on määratud luu, kuvatakse uus lahter, kus on võimalik määrata sihtmärgi täpne asukoht luu peal.

Vertex Group (tipugrupp)

Kui sihtmärgiks on võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp (Vertex Group).

Rest Length (puhkeasendi pikkus)

See numbriväli määrab ära valdaja ning sihtmärgi vahelise kauguse puhkeasendis, st kauguse, mille korral valdajat ei deformeerita (ei venitata).

Reset (algväärtusta)

See nupp algseadistab puhkeasendi pikkuse (Rest Length) väärtuse vastavusse valdaja ja sihtmärgi tegeliku vahemaaga (nendevaheline kaugus enne piiraja rakendamist).

Volume Variation (mahumuutus)

See numbriväli määrab "mahuvariatsioonide" suuruse proportsioonis venituse suurusega. Pane tähele, et väärtus **0.0** ei ole lubatud; mahuvariatsioonide tühistamiseks kasuta nuppu None (puudub).

Volume (maht)

Need nupud määravad, kas objekti mõõtkava teisendatakse mööda X- ja/või Z-telge, et säilitada tema Y-telje suunas venitamise puhul virtuaalset mahtu.

Kui aktiivne on nupp None (puudub), on mahu muutmine välja lülitatud.

Plane (tasand)

Need nupud on analoogilised [järgimise piiraja](#) nuppudega Up (üles): nad määravad, kas objekti Y-telje orienteerimisel sihtmärgi suunas peaks kas X- või Z-telg joondatama nii hästi kui võimalik globaalse Z-teljega.


Järgimise piiraja (*Track To*)

Kirjeldus

Järgimise piiraja (*Track To*) muudab oma valdajale pööret nii, et selle määratud telg (*To*) oleks alati suunatud sihtmärgi poole ja teine telg (*Up*) oleks püsivalt ja võimalikult hästi joondatud globaalse Z-teljega (vaikimisi). See piiraja on samane kolmedimensioonilise "tahveljärgimisega" (vaata märkust allpool).

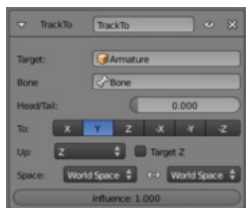
See on järgimiseks kõige eelistatum piiraja, kuna tema järgimismehhanismi on kõige kergem juhtida.

See piiraja on mitmel viisil tihedalt seotud [pöördkinemaatika piirajaga](#). Järgimise piiraja on väga oluline skeleti loomisel, ja sa peaksid kindlasti lugema ja aru saama [2.49 BSoD järgimise õppetükist](#), mis keskendub nende kahe piiraja kasutamisele.

 **Tahveljärgimine**

Tahveõ (*billboard*) omab reaalaraja arvutigraafika programmeerimises (näiteks videomängud) spetsiifilist tähendust: see on tasapinnaline objekt, mis on alati suunatud kaamera poole (tegemist on tõepoolest "järgijaga", mille "sihtmärgiks" on kaamera). Peamiselt kasutatakse seda puu- või udutekstuuride abivahendina: kui need ei oleks pidevalt kaamera suunas pööratud, näeksite tihtipeale puid külgsuunas kokkupressituna või udu nägemas välja nagu Napoleoni kook - mis oleks päris naljakas, ent mitte eriti loomutruu.

Valikud



Järgimise piiraja paneel
Track To

Targets (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgiks (Target) on skelett, kuvatakse uus lahter, kus on võimalik sihtmärgiks valida konkreetne luu.

Head/Tail (pea/saba)

Kui sihtmärgiks on määratud luu, kuvatakse uus lahter, kus on võimalik määrata sihtmärgi täpne asukoht luu peal.

Vertex Group (tipugrupp)

Kui sihtmärgiks on võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp (Vertex Group).

To (suund)

Sihtmärgi järgiv telg (vaikesättena Y), st valdaja lokaalne koordinaattalg, mis on alati suunatud sihtmärgi poole. Negatiivsed väärtused panevad järgiva telje osutama sihtmärgist eemale.

Up (üles)

Kohalik püsttelg (vaikesättena Z) ehk valdaja telg, mis joondatakse globaalse Z-teljega nii hästi kui vähegi võimalik (või sihtmärgi Z-teljega, kui nupp Target Z (sihtmärgi Z) on aktiveeritud).

Target Z (sihtmärgi Z)

Vaikesättena on valdaja püsttelg (Up) nii hästi kui võimalik joondatud globaalse Z-teljega. Selle nupu aktiveerimisel joondatakse püsttelg (Up) võimalikult täpselt sihtmärgi kohaliku Z-teljega.

Space (koordinaatsüsteem)

Selle piiraja puhul on võimalik valida, millises koordinaadistikus tema valdaja ning sihtmärgi teisendusomadusi arvutatakse.



Kui määrata piiraja järgiv telg (To) ja püsttelg (Up) samaks, siis piiraja ei toimi ning on "punases olekus".

Tegevuse piiraja (*Action*)

Kirjeldus

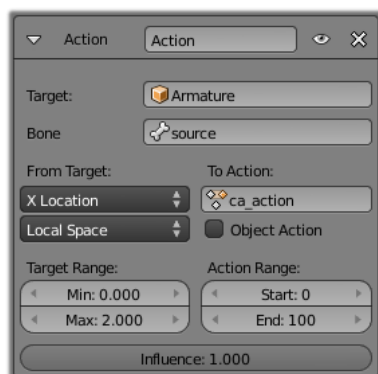
Tegevuse (Action) piiraja on võimas tööriist, kuid temast on omajagu keeruline aru saada. Põhijoontes võimaldab tegevuspiiraja omistada valdajale mingi tegevuse, kasutades sihtmärgi ühte teisendusomadust, määramaks valdaja asukohta (või õigemini aega) selles tegevuses.

Jah, ma mõistan, et see pole kuigi arusaadav. Ennekõike tuleb aru saada, mida tähendab [tegevus](#) (*Action*) Blenderis. Tegevuse piiraja (Action) on väga sarnane [IPO-juhtijatele](#) selle vahega, et tegevuse piiraja kasutab tervet tegevust (mitut sama tüüpi IPO-kõverat) ning IPO-juhtijatele kontrollivad ainult üht oma valdaja IPO-kõverat.

Pane tähele, et ehkki piiraja aktsepteerib võre (Mesh) tüüpi tegevust, töötavad tegelikult ainult objekti (Object), poosi (Pose) ja piiraja (Constraint) tegevustüübid, kuna piirajad saavad mõjutada ainult objektide ja luude teisendusomadusi, mitte võrede kuju.

Näiteks oletame, et defineeritud on objekti (Object) tüüpi tegevus (see võib olla määratud mistahes objektile või ka mitte millelegi) ning seejärel seostatud tegevuse piiraja (Action) abil valdajaga nii, et sihtmärgi liigutamisel X telje vahemikus $[0.0, 2.0]$ toimub valdaja tegevus kaadrite vahemikus $[0, 100]$. See tähendab, et kui sihtmärgi X väärtus on **0.0**, siis valdaja olek on sama mis kaadris **0**; kui sihtmärgi X väärtus on **1.0**, on valdaja olek vastav kaadrile **50** jne.

Valikud



Tegevuse piiraja paneel Action

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ja ei tööta ilma selleta (on punases olekus).

Action (tegevus)

Võimaldab valida soovitud tegevuse nime.



Kui selles väljas pole tegevust määratud, siis piiraja ilmselgelt ei tööta, ehkki ta ei pruugi olla punases olekus (kasutajaliidese värskendusprobleemide tõttu...)

Transform Channel (teisenduskanal)

See rippmenüü määrab, millist teisendusomadust (asukoht, pööre või mõõtkava piki või ümber määratud telje) kasutatakse "tegevuse juhina".

Action Length Start/End (tegevuse kestus algus/lõpp)

Seostatava tegevuse algus- ja lõppkaader.

Pea meeles, et:

- Need väärtused peavad olema rangelt positiivsed.
- Vaikesättena on mõlemad väärtused **0**, lülitades sellega tegevuse seostamise välja (valdaja omadused loetakse seostatud tegevuse kaadrist **0**).

Target Range Min/Max (sihtmärgi vahemik min/max)

Tegevust määrava sihtmärgi teisendusomaduse ülemine ja alumine piir.

Vaikesättena on mõlemad väärtused **0.0**



Kahjuks kehtivad siinkohal teatud piirangud:

- Kasutades tegevuse "juhina" pööret, teisendatakse need väärtused vahemikku $[-180.0^\circ, 180.0^\circ]$.
- Kasutades tegevuse "juhina" mõõtkava, pole võimalik kasutada negatiivseid väärtuseid.

Space (koordinaatsüsteem)

Selle piiraja puhul on võimalik valida, millises koordinaatsüsteemis tema sihtmärgi teisendusomadusi arvutatakse.

Märkused

- Kui seostatud tegevus mõjutab mõnda asukoha omadustest, liidetakse piiraja arvutatud väärtusele valdaja olemasolev asukoht (samamoodi nagu nupu Offset (nihe) aktiveerimisel [asukoha kopeerija \(Copy Location\) puhul](#)).
- Kui seostatud tegevus mõjutab mõnda mõõtkava omadustest, korrutatakse piiraja arvutatud väärtus valdaja olemasoleva mõõtkavaga.
- Kui seostatud tegevus mõjutab mõnda pöörde omadustest, kirjutatakse valdaja olemasolev pööre piiraja arvutatud väärtusega üle.
- Erinevalt tavalisest on siin võimalik sisestada väljale Start (algus) suurem väärtus, kui on väljal End (lõpp), samuti väljale Min

(miinimum) suurema väärtus kui väljale Max (maksimum). See pöörab tegevuse ümber (st seda "esitatakse" tagurpidi), seda loomulikult ainult sel juhul, kui mõlemad väärtuste paarid pole ümber pööratud.

- Kasutades tegevusena piirajat (*constraint*), määratakse see, millistele valdaja piirajatele tegevus rakendub, piiraja *kanalite nimede* poolt. Näiteks kui sul on piiraja kanal nimega "trackto_empty1", seostatakse tema animeeritud mõju (Influence) ja/või Head/Tail (pea/saba) väärtused (ainsad, mida saab animeerida) valdaja piirajaga "trackto_empty1".
- Samaselt, kasutades tegevusena poosi (mis on ilmselt mõistlik ja töötab ainult siis, kui piirajat rakendatakse luudele), määrab luu nimi selle, millist tegevuse *luukanali* nime kasutada. Näiteks kui luu nimi on "arm" (õlavars), kasutab ta tegevuse luukanalit "arm" ja ainult seda. Kahjuks ei ole võimalik poosi (*Pose Action*) rakendada tervele skeletile (et mõjutada tegevuse kõiki animeeritud luid korraga)...
- Pane lisaks tähele, et sa saad kasutada poosi tegevuse andmeblokkide loomiseks ja muutmiseks [pooside kogu](#)... Pea ainult meeles, et iga kaadri kohta on üks poos.

Alluvuse piiraja (*Child Of*)

Kirjeldus

Alluvuse piiraja (Child Of Constraint) on tavalise objektidevahelise alluvussuhte (mida saab 3D-vaates luua kiirklahviga CtrlP) piirajaversioon.

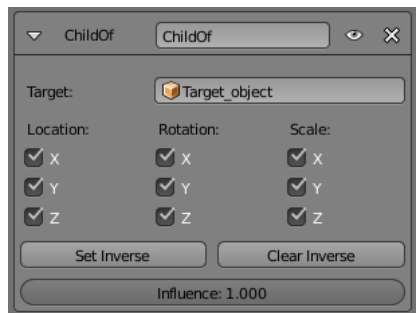
Piirajaga alluvussuhete kehtestamisel on traditsioonilise meetodi ees mitmeid eeliseid ja võlusid:

- Ühel objektil võib olla mitu ülemobjekti (mille mõju on kaalutud vastavalt liuguri Influence (mõju) väärtusele).
- Nagu igal teisel piirajal, saab alluvuspiirajal määrata võtmekaadrede (st animeerida) mõju (Influence) väärtusele ja seega näiteks allutada piiraja valdajaobjekt animatsiooni alguses täielikult ühele ja animatsiooni lõpus teisele ülemobjektile.



Ära aja neid "põhilisi" objektide alluvussuhteid segamini [luuahelate](#) alluvussuhetega skeletis. Seda piirajat on võimalik kasutada objekti allutamiseks luudele (niinimetatud "[objekti nahaga katmine](#)") või isegi luude allutamiseks teistele luudele. Ent ära püüa kasutada seda luuahelate defineerimiseks.

Valikud



Alluvuse piiraja paneel Child Of

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus). Kui sihtmärgiks on määratud skelett või võre, kuvatakse uus lahter, kus on võimalik valida luu või tipugrupp.

Location X, Y, Z (asukoht X, Y, Z)

Need nupud panevad ülemobjekti mõjutama alluva asukohta piki määratud telge.

Rotation X, Y, Z (pööre X, Y, Z)

Need nupud panevad ülemobjekti mõjutama alluva pööret ümber vastava telje.

Scale X, Y, Z (mõõtkava X, Y, Z)

Need nupud panevad ülemobjekti mõjutama alluva mõõtkava piki vastavat telge.

Set Inverse (määra pöördteisendus)

Kui sa allutad piiraja valdajaobjekti sihtmärgile, saab sihtmärgist vaikimisi tema kohaliku koordinaatsüsteemi kese. See tähendab, et valdaja asukoht, pööret ja mõõtkava nihutatakse sihtmärgi samade teisendusomaduste väärtuste võrra. Teisisõnu, kui sa piiraja valdajaobjekti sihtmärgile allutad, teisendatakse teda.

See ei pruugi olla soovitud tulemus! Seega, kui sa tahad jätta valdaja teisenduse samaks, mis tal oli enne sihtmärgile allutamist, vajuta nuppu Set Inverse (määra pöördteisendus).

Clear Inverse (kustuta pöördteisendus)

See nupp tühistab eelmise nupu mõju ning taastab valdaja/alluva vaikeseisundi sihtmärgi suhtes.

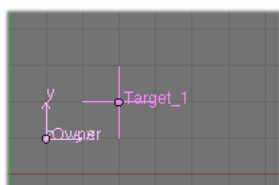
Näpunäited

Kui sa lood selle piirajaga uue hierarhilise suhte, tuleb üldjuhul pärast ülemobjekti määramist vajutada nuppu Set Offset (määra nihe). Nagu eelpool öeldud, tühistab see kõik soovimatud ülemobjektist tulenevad teisendused ning valdaja asend/pööre/mõõtkava muutuvad samasugusteks, nagu nad olid enne piiraja rakendamist. Pane tähele, et valik Set Inverse (määra pöördteisendus) tuleks rakendada siis, kui kõik ülejäänud piirajad on vastava alluvuspiiraja (Child Of) suhtes välja lülitatud (nende mõju (Influence) väärtus on 0.0) ja enne sihtmärgi/ülemobjekti teisendamist (loe ka allolevat näidet).

Kõik lülitid, mis määravad, millised sihtmärgi/ülemobjekti individuaalsed teisendusomadused mõjuvad valdajale, tasub jätta sisse lülitatuks või siis lülitada nad välja kolmekaupa: kõik sama teisendustüübi omad korraga.

Kui selle piiraja kasutamisel on kõik kanalid aktiivsed, kasutab ta alluva mõjutamisel lihtsat maatriksi korrutamist ega jaga seda eraldi nihke, pööre ja mõõtkava alammaatriksiteks. See tagab, et kõik teisendused rakendatakse korrektselt ka siis, kui ülemobjekti teisendus sisaldab korraga pöördeid ja ebaühtlast mõõtkava muutmist.

Näide



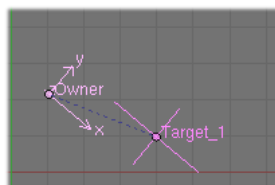
1. Ilma piirajata

Pane tähele valdaja tühiobjekti asukohta – **1.0 BU** piki X ja Y telgi.



2. Lisati alluvuse piiraja (Child Of)

On näha, et valdaja tühiobjekt on nüüd **1.0 BU** kaugusel sihtmärgiks olevast nullobjektist Target_1 piki X ja Y telgi.



3. Määratud on nihe

Vajutatud on nuppu *Set Inverse* ning valdaja on naasnud algpositsioonile.



4. Teisendatud sihtmärk/ülemobjekt

Target_1 on nihutatud XY-tasandile, pööratud ümber Z-telje ning selle mõõtkava on teisendatud piki oma kohalikku X-telge.



5. Nihke tühistamine

Vajutatud on nuppu *Clear Inverse* ja valdaja on jälle täielikult sihtmärgi Target_1 kontrolli all.

6. Nihe uuesti määratud

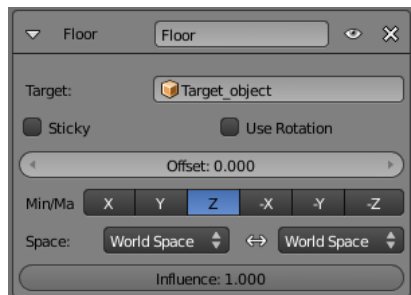
Set Offset nuppu on uuesti vajutatud. Nagu näha, on tulemus erinev kui punktis "Teisendatud sihtmärk/ülemobjekt". Nagu eespool öeldud, tuleks nuppu *Set Inverse* vajutada ainult üks kord ja seda enne sihtmärgile/ülemobjektile teisenduste rakendamist.

Põranda piiraja (*Floor*)

Kirjeldus

Põranda piiraja (*Floor*) võimaldab sul kasutada sihtmärgi asukohta (ja soovi korral ka pööret) määramaks tasandit, mille üks külg on "keeluala", kuhu valdaja ei saa liikuda. See pind võib olla ükskõik kuidas orienteeritud. Teisisõnu loob see piiraja põranda (või seina või lae...!)

Valikud



Põranda piiraja paneel Floor

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Bone (luu)

Kui sihtmärgiks (Target) on skelett, kuvatakse uus lahter, kus on võimalik sihtmärgiks valida konkreetne luu.

Vertex Group (tipugrupp)

Kui sihtmärgiks on võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp (Vertex Group).

Sticky (kleepuv)

See nupp muudab valdaja pärast "põranda" tasandi puudutamist liikumatuks (ta ei saa mööda pinda libiseda). See on suurepärane abi kõndimis- ning jooksuanimatsioonide tegemiseks!

Use Rotation (kasuta pööret)

See nupp paneb piiraja arvestama sihtmärgi pööret. Nii saab "põranda" tasandile anda mistahes suuna, mitte ainult globaalse XY, XZ ja YZ tasanditega paralleelse.

Offset (nihe)

See numbriväli võimaldab "põranda" tasandit nihutada sihtmärgi keskpunktist määratud arvu Blenderi ühikute (BU) võrra. Seda võib kasutada näiteks kompenseerimaks kaugust jalaluust jala pinna võreni.

Max/Min (maksimum/miinimum)

Need nupud (vastastikku üksteist välistavad) näitavad, milline tasand määratakse "põrandaks". Nuppude nimed vastavad tasandite *normaalidele* (näiteks Z tähistab "XY tasandit" jne). Vaikesättena on need normaaliid joondatud globaalsete telgedega. Kui aga pöörde kasutamine (Use Rotation) on sisse lülitatud (vaata eestpoolt), joonduvad nad sihtmärgi *kohalike koordinaattelgedega*. Kuna see piiraja ei määra ainult läbimatut tasandit, vaid ka tasandi külje, mis on valdaja jaoks keeluala, on seda külge võimalik valida, aktiveerides kas positiivse või negatiivse normaaliid... Näiteks vaikesättena on see Z, mis paneb valdaja püsima positiivsete Z-koordinaatide piires.

Space (koordinaatsüsteem)

Selle piiraja puhul on võimalik valida, millises koordinaadistikus tema valdaja ning sihtmärgi teisendusomadusi arvutatakse.

Raja järgimise piiraja (*FollowPath*)

Kirjeldus

Raja järgimise piiraja (Follow Path) asetab valdaja sihtobjekti *kõverale* ning paneb ta mööda seda kõverat (või rada) liikuma. Juhul kui kõvera järgimise valik Follow Curve on aktiivne, muudab ta ka valdaja pööret, järgides kõvera pöördeid.

Valdaja teisendus arvutatakse alati globaalses (maailma) koordinaadistikus:

- Tema asukoht (nagu näidatud teisendusomaduste (Transform Properties) paneelil, kiirklahv N), määrab nihke tema normaalse asukoha suhtes rajal. Näiteks kui valdaja asukoht on (1.0, 1.0, 0.0), paikneb ta piiraja rakendamisel piki X ja Y telgi ühe Blenderi ühiku (BU) kaugusel tema normaalsest asukohast kõveral. Seetõttu tuleb valdaja asetamiseks *täpselt* sihtmärgiks olevale rajale tema asukoht algseadistada kiirklahviga (AltG)!
- Asukoha nihet mõjutab proportsionaalselt ka *sihtkõvera mõõtkava*. Kui valdaja on samasuguse nihkega (1.0, 1.0, 0.0), kuid sihtmärgkõvera mõõtkava on (2.0, 1.0, 1.0) nihutatakse valdajat kaks BU-d mööda X telge ning üks BU mööda Y telge
- Kui kõvera jälgimine (Curve Follow) on aktiivne, muudetakse ka tema pööret vastavalt kõvera pöördele (seega, kui sa tahad, et sinu objekti Y telg ühtiks piiraja rakendamisel kõvera suunaga, peab ta ilma piirajata puhkeolekus ühtima globaalse Y teljega). Siinkohal võib samuti olla kasulik valdaja pöörde algseadistamisest kiirklahviga (AltR)...

Valdaja liikumist mööda sihtmärgi kõverat/rada saab juhtida kahel erineval viisil:

- Kõige lihtsam on määrata liikumise kaadrite arv raja animatsiooni (Path Animation) paneelil, kõvera (Curve) menüüs numbrivälja Frames (kaadreid) abil ja alguskaader piiraja valikuga Offset (nihe) (vaikesätted on alguskaader: **1** [= nihe **0**]), kestus: **100**).
- Teine, palju täpsem ja võimalusterohkem viis on määrata sihtmärgi (Target) trajektoori kiiruse (Speed) interpolatsioonikõver (graafikuredaktoris Graph Editor, kiirklahviga ⇧ ShiftF6). Stardipositsioon rajal vastab IPO väärtusele **0.0** ning lõppväärtusele **1.0**. Sellega on võimalik mõjutada alguskaadrit, liikumise kiirust, lõpukaadrit ning isegi sundida objekti liikuma edasi ja tagasi mööda rada! Loe IPO-kõverate kohta pikemalt [animatsiooni peatükist](#).
- Kui sa ei taha, et valdaja piki trajektoori liiguks, võid sa anda sihtmärgkõverale konstantse kiiruse (Speed) IPO (selle väärtus määrab siis valdaja positsiooni trajektoiril).

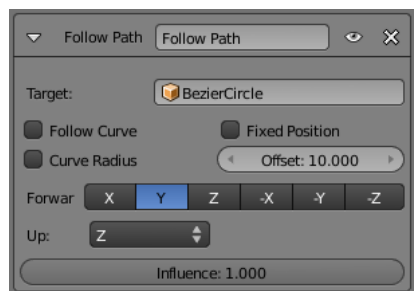
Raja järgimine Follow Path töötab hästi koos [lukustatud trajektoori piirajaga \(Locked Track\)](#). Hea näide on mööda rada liikuv kaamera. Et juhtida kaamera küljelt-küljele pöördenuka piki rada liikudes, võid sa kasutada lukustatud trajektoori piirajat (Locked Track) ning sihtmärgobjekti, mis määrab kaamera püsttelje.

Raja järgimise (Follow Path) ja klammerduspiiraja (Clamp To)

Ära neid piirajaid segamini aja. Mõlemad piiravad nende valdaja asukohta kõveral, ent raja järgimise piiraja (Follow Path) töötab ainult "animatsioonis", niivõrd kui valdaja asukoha kõveral määrab animatsiooni aeg (praegune kaader), samas kui klammerduspiiraja ([Clamp To](#)) määrab valdaja positsiooni kõveral, kasutades mõnda tema asukohaseadete väärtustest.

Pane tähele, et sul on vaja ka raja jaoks võtmekaadri aega (*Keyframe Evaluation Time*). Vali rada ning mine selle omaduste aknasse, määra raja stseeni üldkaadriks raja esimene kaader (näiteks 1), määra väljal *Evaluation Time* (arvutusaeg) raja esimese kaadri väärtuse arvutamise aeg (näiteks 1) ja loo hiire paremklopsuga sellel väljal uus võtmekaader (*Keyframe*). Seejärel määra stseeni üldkaadriks raja viimane kaader (näiteks 100), määra väljal *Evaluation Time* (arvutusaeg) raja viimase kaadri arvutamise aeg (näiteks 100), klõpsa parema hiirenupuga sellel väljal ning loo uus võtmekaader (*Keyframe*).

Valikud



Raja järgimise paneel Follow Path

Target (sihtmärk)

See piiraja kasutab üht sihtmärki, mis peab olema kõver, ning ilma selleleta piiraja ei tööta (on punases olekus).

Curve Radius (kõvera raadius)

Objektide mõõtkava teisendatakse vastavalt kõvera raadiusele

Fixed Position (fikseeritud asend)

Objekt püsib lukustatuna teatud kõvera punktis, hoolimata (animatsiooni) ajast

Offset (nihe)

Kaadrite arv, mille võrra nihutada piiraja "algust" võrreldes raja poolt määratud "animatsiooniga" (vaikesättena algab kaadrist number 1).

Follow Curve (järgi kõverat)

Kui see valik on välja lülitatud, ei sõltu valdaja pööre kõvera omadustest. Vastasel juhul mõjutavad teda järgmised valikud:

Forward (edasi)

Objekti telg, mis peab joonduma rajal edasi liikumise suunaga (st kõvera puutujaga valdaja asukohas).

Up (püsttelg)

Objekti telg, mis peab joonduma (nii palju, kui võimalik) maailma Z-teljega.

Kui see valik on aktiveeritud, siis on valdaja käitumisel tegelikult mõned ühised jooned tema käitumisega [lukustatud trajektoori \(Locked Track\)](#) piiraja rakendamisel: rada käitub samamoodi kui "telg" ja maailma Z-telg samamoodi kui "magnet".

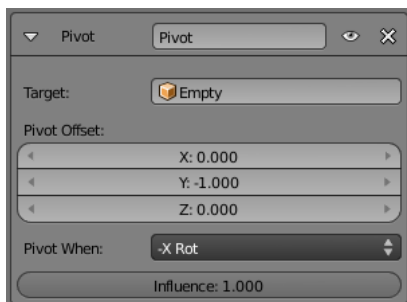
Pöörlemiskeskme piiraja (*Pivot*)

Kirjeldus

Pöörlemiskeskme piiraja (Pivot) laseb valdajal pöörelda ümber sihtmärgi.

See loodi esialgu jalgade skeleti jaoks.

Valikud



Pöörlemiskeskme piiraja paneel Pivot

Target (sihtmärk)

Pöörlemise keskpunktina kasutatav objekt.

Bone (luu)

Kui sihtmärgiks on valitud skelett, kuvatakse uus lahter, kus on võimalik sihtmärgiks valida konkreetne luu.

Head/Tail (pea/saba)

Kui sihtmärgiks on määratud luu, kuvatakse uus lahter, kus on võimalik määrata sihtmärgi täpne asukoht luu peal.

Vertex Group (tipugrupp)

Kui sihtmärgiks on võre, kuvatakse uus lahter, kus on võimalik sihtmärgiks määrata konkreetne tipugrupp (Vertex Group).

Pivot Offset (keskpunkti nihe)

Pöörlemiskeskme nihe sihtmärgi suhtes

Pivot When (kasuta keset)

Always, Z Rot, Y Rot (alati, Z pööre, Y pööre)

Näide

[\[video link\]](#)

Loe lisaks

- [Blender Artists Forum: Head-Tail pivot Constrain proposal \(with Video and .blend\)](#) - foorumivestlus, kus see piiraja kõigepealt ette pandi.

Jäiga keha liigendi piiraja (*Rigid Body Joint*)

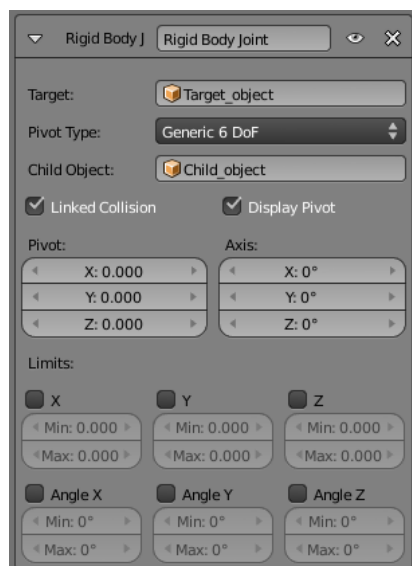
Kirjeldus

Jäiga keha liigendi piiraja (Rigid Body Joint) on väga eriline. Peamiselt kasutatakse seda Blenderi mängumootorifüüsikas valdaja ja sihtmärgi vahelise liigendi simuleerimiseks – kas tavalise hinge, kuulliigendi või üldise kuue vabadusastmega liigendi.

Kinnituskoht ning teljed on alati määratud valdaja suhtes. Sihtmärk liigub justkui kepi otsas, mille üks ots kinnitub tema keskpunkti ja teine ots pöörleb ümber liigendi telje/keskpunkti...

Tavalistes staatilistes või animatsiooniprojektides pole selle piirajaga eriti midagi teha. Siiski võib selle tulemusi kasutada ka väljaspool mängumootorit, salvestades animatsiooni menüüst Game » Record Animation (salvesta animatsioon) (kasutajaeelistuste (User Preferences) akna peamenüüs; vaata lähemalt [jäikade kehade peatükist](#)).

Valikud



Jäiga keha liigendi paneel Rigid Body Joint

Target (sihtmärk)

See piiraja kasutab üht sihtmärki ning ilma selleta ei tööta (on punases olekus).

Joint Type (liigendi tüüp)

Ball (kuul)

Töötab perfektse kuulliigendina, st lubab pöördeid kõikide telgede ümber nagu õlaliiges.

Hinge (hing)

Töötab ühetasapinnaliselt nagu küünarnukk – valdaja ja sihtmärk võivad pöörduda ainult liigendi X-telje ümber.

Limits (piirid)

X-telje pöördenurga piirid

Cone Twist (koonuse vääne)

Nagu kuulliigend (Ball), millele on lisatud maksimaalse koonuse nurga ja väände piirangud.

Limits (piirid)

Nurga piirangud

Generic 6DOF (kuue vabadusastmega)

Töötab nagu kuulliigend (Ball), kuid sihtmärgi kaugus liigendist võib vaikimisi muutuda (kuus vabadusastet tähendavad pöördeid ja nihkeid ümber/piki kolme koordinaattelge).

Tegelikult vaikesätete järgi siinkohal enam otseselt liigendit ei olegi, ent seadete abil on võimalik vabadusastmeid piirata:

Limits (piirid)

Nihke ja nurga piirid vastavatel (pöörlemiskeskme) telgedel väljendatuna vastavalt Blenderi ühikutes ja kraadides.

Child Object (alluv)

Tavaliselt jätta tühjaks. Saad selle väärtuse tühistada, vajutades väljale parema hiire klahviga ja valides *Reset to Default Value* (taasta algne väärtus).

Linked Collision (seotud pörge)

See valik tühistab mängumootoris valdaja ja sihtmärgi vahelise pörge testimise.

Display Pivot (näita liigendit)

Aktiveeritult joonistab 3D-vaates liigendi pöördepunkti. See on väga kasulik eriti kuueastmelise liigendi (Generic 6DOF) puhul!

Pivot (pöörlemiskese)

Need kolm numbrivälja lubavad muuta liigendipunkti asukohta *valdaja koordinaadistikus*.

Axis (telg)

Need kolm numbrivälja lubavad pöörata liigendipunkti *valdaja koordinaadistikus*.

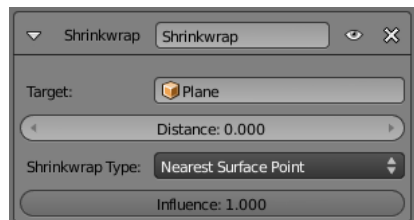
Sobitamispiraja (*Shrinkwrap*)

Kirjeldus

Sobitamispiraja (Shrinkwrap constraint) on "objekti tasandi vaste" [sobitamistöötlejale](#). See piiraja kahandab valdaja asukoha sihtmärgi *pinnale*.

See tähendab, et sihtmärgil *peab* olema pind. Tegelikult on see piiraja veel nõudlikum ja lubab sihtmärgina kasutada ainult võresid. Seetõttu on sobitamispiraja valik nähtav ainult piiraja lisamisel objektile menüü *Add Constraint to Active Object* (lisa aktiivsele objektile piiraja) (kiirklahv CtrlAltC) või vastava luu menüü kaudu, kui valitud mitteaktiivseks objektiks on võre.

Valikud



Sobituspiiraja paneel Shrinkwrap

Target (sihtmärk)

See piiraja kasutab üht sihtmärki, mis peab olema võre, ning ilma selleta piiraja ei tööta (on punases olekus).

Distance (kaugus)

See numbriväli määrab valdaja nihke sihtmärgi pinnal asuva "kokkutõmmatud" asukoha suhtes.

Positiivsed väärtused asetavad valdaja sihtmärgist "väljaspoole", negatiivsed sihtmärgi "sisse".

Nihe rakendatakse piki sirgjoont, mis ühendab valdaja esialgset asukohta (enne piiraja rakendamist) ning arvutatud asukohta sihtmärgi pinnal.

Shrinkwrap Type (sobitamise tüüp)

Sellest rippmenüüst saab valida meetodi, mille põhjal arvutatakse valdaja keskpäiga asukoht sihtmärgi pinnal. Valikuid on kolm:

Nearest Surface Point (lähim pinnapunkt)

Valitud punkt sihtmärgi pinnal on kõige lähemal valdaja algsele asukohale. See on kõige enam kasutatav variant, samuti vaikesäte.

Projection (projektsioon)

Sihtmärgi pinnalaotuse punkt valitakse, projitseerides valdaja keskpunkti piki antud telge.

See telg valitakse kolme lülitiga (X, Y ja Z), mis kuvatakse pärast selle sobitustüübi valimist. See tähendab, et projektsioonitelg võib joonduda ühega globaalsetest telgedest või nende mediaanidega (XY, XZ, YZ) või kõigi kolme mediaaniga (XYZ).

Kui valdaja keskpunkt ei läbi piki valitud telge projitseerides sihtmärgi pinda, jääb valdaja asukoht muutumatuks.

Nearest Vertex (lähim tipp)

See meetod on väga sarnane *lähima pinnapunkti* meetodiga, ent valib valdaja sobitatud keskpunkti asukohaks sihtmärgi lähima tipu.

Animeerimine

Animeerimine tähendab panna objektide kuju või asukoht ajas muutuma. Objekte saab animeerida mitmel viisil:

Tervikliku objekti liigutamine

positsiooni, asendi või suuruse muutmine ajas;

Kuju muutmine

kontrollpunktide või tippude animeerimine;

Tegelaste animeerimine skeleti abil

objekti deformeerimine sõrestiku sisemuses asuvate luude liigutamise abil. See on keerukas ja paindlik meetod, mille abil saab panna tegelaskujusid kõndima ja hüppama.

Selles peatükis me käsitleme kahte esimest, ent siin toodud põhitõed on vajalikud ka järgmiste peatükkide mõistmisel.

3D-objektide liigutamiseks kasutatakse animatsiooniprogrammides tavaliselt kolme meetodit:

Võtmekaadrid

Iga ajahetke (kaadri) jaoks salvestatakse objekti täielik asend. Animatsioon luuakse objekte kaadrite vahel sujuvalt interpoleerides. Selle meetodi eeliseks on hea visuaalne ülevaade animatsiooni üksik sammudest. Animaator saab liikuda ühest positsioonist järgmisse, muuta varem tehtud positsioone või neid ajas liigutada.

Animatsioonikõverad

Kõverad interpoleeritakse võtmekaadritest. Neid saab määrata eraldi kõigi asukoha, pöörde ja skaala XYZ-komponentide või ükskõik milliste muude Blenderi atribuutide jaoks. Nad kujutavad endast liikumise graafikut, kus aega näidatakse horisontaal- ja väärtust vertikaalteljel. Selle meetodi eeliseks on täpne kontroll liikumise üle igal ajahetkel.

Teekond

3D-ruumi joonistatakse kõverjoon ja objekt pannakse läbima teekonda vastavalt sellel määratud ajapunktiledele.

Blenderis on kaks esimest süsteemi täielikult kokku liidetud ühiseks [F-kõverate süsteemiks](#) (*F-Curve system*).

Alates Blenderi versioonist 2.5x on võimalik animeerida kõike. Varem oli võimalik määrata võtmekaadreid ainult osadele andmeplokkidele. Nüüd saavad kasutajad animeerida peaaegu kõiki andmetüüpe, millel on mitu erinevat võimalikku väärtust.

Peatükid

Üldpõhimõtted ja tööriistad

[Võtmekaadrid](#)
[Animatsiooniredaktorid](#)
[Ajatelje kasutamine](#)
[Markerid](#)

Graafikuredaktor

[Graafikuredaktori kasutamine](#)
[F-kõverad](#)
[Kõverate muutmine](#)
[F-kõvera töötlejad](#)

Tegevusredaktor

[Tegevused](#)
[Tegevuste loomine](#)

Animatsioonitehnikad

[Piirajad](#)
[Objekti liigutamine piki joont](#)
[Objekti kihtide muutmine](#)
[Mängumootori füüsika salvestamine](#)

Deformatsiooni animeerimine

[Moonutusmeetodid](#)
[Võtmevormid \(*Shape Keys*\)](#)
[Absoluutsed võtmevormid](#)
[Deformeerimine sõrestiku abil](#)
[Deformeerimine haakidega](#)

Vaata ka [Haagid](#) - siin kasutatakse töötlejat võre kuju muutmiseks. Umbes nagu õngekonksuga sõrestikku sikutades. Töötab võtmevormides käsitletud põhimõttel.

Juhtijad

[Juhtijad \(*Drivers*\)](#)
[Juhtivad võtmevormid \(*Driven Shape Keys*\)](#)

Hea koht animatsiooni õppimist alustada on [BSoD tegelaskuju animatsiooni sissejuhatav õppetükk](#). Isegi kui sa pole kunagi varem Blenderit kasutanud.

Ajatelg (*Timeline*)

Ajatelje aken, mille tunnuseks on kellaga ikoon, on vaikumisi nähtav ekraani allääres.


Ajatelg ei ole tegelikult "redaktor", vaid sarnaneb rohkem ülevaateaknaga *Outliner* – see on "informatsiooni" aken, milles on mõned piiratud muutmise võimalused.

Siin kajastub sinu stseeni animatsiooni üldpilt: mis on praegune ajahetk (kas kaadrites või sekundites), kus asuvad aktiivse objekti võtmekaadrid, millised on sinu animatsiooni algus- ja lõppkaadrid, markerid, jne.

Sellel on videomagnetofonilikud seadenupud, mille abil saad animatsiooni maha mängida, muuta praegust kaadrit ja kaadrite vahemikku ning kerida võtmekaadrite vahel.

Ajatelje elemendid

Praegune kaader

Praegust kaadrit tähistab paks roheline vertikaaljoon (seda kutsutakse "aja kursoriks" ning galeriis *Ajatelje aken* asub see kaadris **100**). Sa saad seda liigutada, vajutades aknas ükskõik millisesse kohta klahviga LMB , samuti saad sa animatsiooni edasi-tagasi kerida, vajutades sama klahvi ja lohistades hiirt. Reaalset kaadri numbrit (või aega sekundites) kuvatakse vajutamise või lohistamise ajal kursori kõrval – ning loomulikult on see alati nähtav "praeguse kaadri" numbriväljal [päises](#).

Võtmekaadrid (*Keyframes*)

Mõnesid aktiivse objekti võtmekaadreid (või aktiivseid [võtmekaadreid](#), jms) kujutatakse vertikaalsete värviliste joontega selles kaadris, kus need asuvad. Ma usun, et sedasi kuvatakse ainult kolme IPO-tüüpi:

- kollane**

Tegevusega mitteseotud *objekti* võtmed (asukoht, pööre jne)

- oranž**

Materjali võtmed (hajusus, läige, peegelduse värvid jne)

- rohekassinine**

[Tegevusi](#) loovad võtmed (alati skeleti luude poosid, kuid võib-olla ka objektide ja kuju IPOd)

Märkus

Mõnede IPO-tüüpide võtmeid ei kuvata üldse (nt *Tekstuuri* või *Piirajate* omasid).

Galeriis ([Ajatelje aken](#)) on kaks "objekti" võtmekaadrit (kaadrid **10** ja **150**), üks "materjali" võtmekaader (kaader **40**) ning mitte ühtegi "tegevuse" võtmekaadrit.

Markerid (*Markers*)

[Markereid](#) kujutatakse väikeste kolmnurkadena, mille lähedal on kirjas nende nimi.

Värvikoodid on:

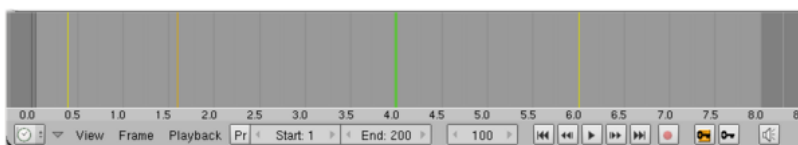
- valge joon, must tekst:** valimata markerid
- kollane joon, valge tekst:** valitud markerid

Vaate muutmine

Ajatelje aken.






Kaadrite kuvamine.



Sekundite kuvamine.

See aken on ühemõõtmeline ja kujutab horisontaalsel teljel ainult sinu stseeni aega. Kuna kaadrid on Blenderi peamine ajaühik, kuvatakse ajatelje alumises servas vaikumisi kaadrite numbreid.

Animatsiooni vahemikku tähistab heledam hall toon (galeriis *Ajatelje aken* on see vahemikus kaader **1** kuni kaader **200**).

See aken töötab nagu iga teine Blenderi "ala": saad seda klahvi MMB  all hoides lohistada (ainult vasakule/paremale, kuna tegemist on ühemõõtmelise aknaga), kombinatsiooni Ctrl MMB  või Wheel  abil suurendada/vähendada jne.

Vaate menüü (*View*)

See menüü määrab, mida sa näed ja kuidas sa seda näed:

Maximize Window(maksimeeri aken, CtrlArrowup)

See tavaline käsk paneb ajatelje akna täitma tervet ekraani. Kui aken on maksimeeritud, siis muutub see menüü sissekannne akende taastamise omaks (*Tile Window*), mille abil saad taastada eelneva suuruse (CtrlArrowdown).

Lock Time to Other Windows (lukusta aeg teistes akendes)

See akendevaheline võimalus mõjutab kõiki "aja" aknaid (st kõiki aknaid, mis kujutavad oma X-teljel aega). Kõik sellised aknad, millel on see võimalus sisse lülitatud, näitavad alati sama "ajavahemikku". Kui sa seda ühes aknas muudad (liikudes X-teljel paremale/vasakule või seda suurendades), värskendatakse automaatselt ka kõiki teisi aknaid.

ViewAll (vaata kõike, ⇧ Home)

See tavapärane võimalus vähendab horisontaalselt vaadet nii, et aknasse mahuks terve animatsiooni vahemik (mille määravad algus- (*Start*) ja lõppkaadrid (*End*)).

Center View(tsentreeri vaade, C)

See tavapärane võimalus tsentreerib vaate praegusele kaadriale.

Jump to Prev Key (hüppa eelmisse võtmekaardrisse, CtrlPageDown)

Jump to Next Key (hüppa järgmisse võtmekaardrisse, CtrlPageUp)

Need käsud panevad su kursori hüppama lähimasse aktiivse objekti eelmisse või järgmisse võtmekaardrisse.

Jump to Prev Marker (hüppa eelmise markeri juurde, PageDown)

Jump to Next Marker (hüppa järgmise markeri juurde, PageUp)

Need käsud panevad su kursori hüppama lähima eelmise või järgmise [markeri](#) juurde.

Only Selected Data Keys (ainult valitud andmete võtmekaadrid)

Ma usun, et kui see on sisse lülitatud, peaks ajatelg kuvama ainult valitud võtmekaadreid. Kuid tegelikult näib see alati kuvavat ainult aktiivse elemendi võtmekaadreid!

ShowSeconds/ShowFrames (näita sekunde/näita kaadreid, T)

Vaikimisi kuvatakse aega kaadrites, nii nagu Blender seda siseselt arvutab. See menüü sisekanne lubab sul kuvada aega ka sekundites (vastavalt stseeni kaadrisagedusele).

Sageli on süžetahvlid tehtud sekundite põhisel. Sekundite kuvamine muudab asjad natuke lihtsamaks, sest sa ei pea kogu korrumist peas tegema.

Play Back Animation (mängi animatsiooni)

Mängib animatsiooni praegusest kaadrist lõpuni ning alustab siis selle kordamist algusest lõpuni, kuni vajutad pausi nuppu (või klahvi Esc).

Kõik aknad, mille tagasimängimise menüü *Playback* (vaata altpoolt) sätted sobivad, näitavad animatsiooni mängimas.

Muutmine

Kaadri menüü (*Frame*)

See menüü tegeleb peamiselt [markeritega](#), kuid selles on veel kaks kasulikku sissekannet:

Set as End (määra lõpp, E)

Set as Start (määra algus, S)

Nagu arvata, määravad need kaks käsku praeguse kaadri kas sinu animatsiooni algus- või lõppkaadriks.

Taasesitus

Taasesituse menüü (*Playback*)

See menüü määrab, kuidas ja kus animatsiooni maha mängitakse.

Continue Physics (jätku füüsikaga)

Tööriista vihje ütleb: "Jätka taasesitluse ajal füüsika simulatsioone, hoolimata kaadri numbrist". Ma ei mõista, mida see tähendab.

Set Frames/Sec (määra kaadreid sekundis)

Selle peale hüppab lahti numbri väli, millega saad määrata uue kaadrisageduse.

Märkus: Tuleta meelde [sissejuhatuses](#) olnud hoiatust selle sätte muutmise kohta pärast animatsiooni loomise alustamist – see aeglustab/kiirendab kõiki olemasolevaid animatsioone.'

Sync Playback to Frames/Sec (sünkroniseeri taasesitus kaadrisagedusega)

Sisseelülitatuna sunnib see taasesitlust soovitava kaadrisagedusega sünkroonis püsima.

Pane tähele, et kui Blenderil on piisavalt võimsust, et arvutada sekundis rohkem kaadreid kui vaja, siis püsib ta ikkagi määratud kaadrisageduse juures. Seega mõjub see säte ainult siis, kui animatsioonid on liiga rasked, et neid reaajas arvutada: vaikimisi renderdab Blender kõik kaadrid, mille tulemusena on taasesitus aeglasem. Selle sätte sisseelülitamisel jätab Blender nii palju kaadreid vahele (st ei arvuta neid), kui on vaja normaalse kaadrisageduse säilitamiseks.

Vaata ka "audio sünkroniseerimise" nuppu päises, millest kirjutatakse [allpool](#).

Ülejäänud sätted tegelevad sellega, millistes aknatüüpides tuleks ajateljes käivitatud mahamängimist kuvada. Mida rohkem aknaid sellega tegeleb, seda rohkem sa protsessori jõudlust vajad.

Sequencer Windows (monteerija aknad)

Sisseelülitamisel näitavad kõik videomonteerija (*Video Sequence Editor*) aknad mahamängimist (ükskõik milline nende kuvamisrežiim ka ei oleks: kas montaaž või eelvaade).

Vaata ka "audio sünkroniseerimise" nuppu päises, millest kirjutatakse [allpool](#).

Image Windows (pildi aknad)

Sisselülitamisel näitavad kõik UV/Pildiredaktori (*UV/Image Editor*) aknad mahamängimist.

Buttons Windows (nuppude aknad)

Sisselülitamisel näitavad kõik nuppude aknad (*Buttons Windows*) mahamängimist (selle abil saad sa näha animeeritud sätete väärtuste arengut).

Animation Windows (animatsiooni aknad)

Sisselülitamisel näitavad kõik animatsiooni aknad (nt IPO-kõverate redaktor (*Ipo Curve Editor*), tegevuseredaktor (*Action Editor*) ja mittelineaarse redaktor (*NLA Editor*)) mahamängimist.

All 3D Windows (kõik 3D-aknad)

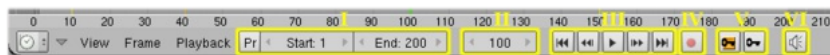
Sisselülitamisel näitavad kõik 3D-vaate (*3D View*) aknad mahamängimist.

Top-Left 3D Window (ülemine vasak 3D-aken)

Sisselülitamisel näitab ainult ülemine vasakpoolseim 3D-vaate aken mahamängimist.

Päise nupud

Päise nupud vastavad enamasti videomagnetofoni omadele:



Ajatelje akna päise nupud.

(I) Animatsiooni vahemik

Esimesed kolm nuppu tegelevad animatsiooni algus- ja lõppkaadritega.

Start/End (algus/lõpp)

Algus- ja lõppkaadrid! Vaata ka alguse ja lõpu määramise kirjeid (*Set as Start/Set as End*) [kaadri menüüs Frame](#) ülalpool.

Pr

Lühend, mis tähistab mahamängimise vahemikku ("Playback Range"). Selle parameetri määravad algus/lõpp (*Start/End*) ning see on "lingitud" stseeni konteksti *Scene* renderduse alamkonteksti *Render* animatsiooni paneelis *Anim* (F10) olevate väärtustega – seega on need väärtused samad. Kui sa aga selle *Pr* nupu sisse lülidad, saad määratleda uue ajutise animatsiooni vahemiku, mis töötab ainult reaajas toimuvate taasesituste puhul (mille algatavad ajatelje "mahamängimise" nupp või kiirvalik AltA). See on väga kasulik, kui pead ainult töötama mõne suure (pika) animatsiooni väikese tükiga!

(II) Praegune kaader

Kolmas numbriline väli näitab ja laseb sul muuta praegust kaadrit (mida kujutab rohelise joone kujuline kursor).

(III) Videomagnetofoni nupud

Nende viie nupu abil saad sa oma animatsioonis navigeerida.

Keskne "mahamängimise" nupp

Alusta taasesitust! Kui mahamängimine toimub, muutub see nupp "pausi" omaks, mis katkestab (peatab) taasesituse.

Esimene ja viimane nupp

Saadavad sind vastavalt algus- ja lõppkaadrisse.

Kaks ülejäänud nuppu

Saadavad sind vastavalt eelmisesse/järgmisesse kuvatavasse võtmekaadrisse.

(IV) Automatiseerimine

Punase punktiga "lindistamise" nupp lülitab sisse võimaluse, mida sageli kutsutakse "automatiseerimiseks": see lisab aktiivse objekti võtmekaadrid ja/või asendab olemasolevaid võtmekaadrid, kui sa seda objekti 3D-vaates teisendad. Pea meeles, et see töötab isegi taasesitamise ajal – mahamängimine katkestatakse teisendamiseks ning pärast teisenduste kinnitamist see jätkub.

Kui sa selle võimaluse sisse lülidad, ilmub nupust paremale automaatsete võtmekaadrite lisamise režiimi rippmenüü *Auto-Keying Mode*, mille abil saab määrata automatiseerimise režiimi:

Replace Keys (asenda võtmekaadrid)

See ainult asendab olemasolevaid võtmekaadrid ja ei loo kunagi uusi. Seetõttu mõjutavad su teisendused animatsiooni ainult siis, kui need toimuvad juba võtmekaadriga varustatud kaadris.

Add/Replace Keys (lisa/asenda võtmekaadrid)

See asendab vajadusel olemasolevaid võtmekaadrid või lisab uusi.

Sa leiad samad sätted (automaatsete võtmekaadrite lisamise nupp *Auto-Keying Enabled*, mis peaks olema võrdväärne "lindistamise" omaga, ning automaatsete võtmekaadrite režiimi rippnimekiri *Auto-Keying Mode*) kasutaja eelistuste akna *User Preferences* muutmismeetodite sakis *Edit Methods*. Kuid tundub, et need sätted ei mõjuta midagi!

Kuid kasutaja eelistuste aknas *User Preferences* on otse nende all kolm lülitit, mis määravad selle, millistele kõveratele see tööriist automaatselt võtmekaadrid lisab:

Available (kõigile)

lisab võtmekaadri kõigile IPO-kõveratele.

Needed (vajalikele)

lisab võtmekaadrid ainult siis, kui neid on vaja (st ainult IPO-kõveratele, mis määravad muudetavaid omadusi).

Use Visual Keying (kasuta visuaalset võtmekaadrite lisamist)

Seda tuleks kasutada objektide või luude puhul, millel on sellised piirajad, mis mõjutavad võtmekaadrite väärtusi. Kui sa näiteks määrad võtmekaadri objektile, millel on asukoha kopeerimise piiraja *Copy Location*, siis tavaliselt määratakse võtmekaadri väärtuseks piiramata asukohta. Selle võimaluse sisselülitamisel määratakse võtmekaadri väärtuseks piiraja poolt mõjutatud asukoht.

Pane tähele, et automatiseerimine töötab ainult teisendatavate omaduste puhul (objektid või luud) ja ainult 3D-vaadetes (st sa ei saa seda näiteks kasutada nuppude aknas *Buttons* materjali värvide animeerimiseks).

(V) Sisestamise ja kustutamise nupud

Kaks "võtmega" nuppu lubavad sul hetke kaadris aktiivse objekti võtmekaadreid sisestada (I, oranž taust) või kustutada (AltI). See ei ole võimalus, mida oleks lihtne mõista või kasutada! See proovib kasutada "suurima ala konteksti" ("alaks" on mõni Blenderi aken). Näiteks kui su suurimaks aknaks on IPO-kõverate redaktor *Ipo Curve Editor*, mis on materjali kontekstis *Material* "context", siis sisestatakse/kustutatakse võtmekaadreid aktiivse materjali IPO-kõverale vastavalt. Kui selleks on mõni 3D-aken, siis näidatakse sulle sama võtmekaadri sisestamise/kustutamise menüüd (*Insert Key/Delete Key*) kui selles 3D-aknas kiirklahvide I/AltI vajutamisel.

(VI) Sünkroniseeri videomonteerija heliga

"Kõlari" nupu sisse lülitamine mõjub põhimõtteliselt samamoodi nagu taasesituse menüü *Playback* kirjade *Sync Playback to Frames/Sec* ja *Sequencer Windows* sisse lülitamine. Mahamängimine kaasab nüüd ka monteerija ning kasutab aja tähistajana selle audio väljundit. See on peamiselt kasulik videomontaaži puhul.

Markerid (*Markers*)

Markereid kasutatakse, et märkida kaadreid, kus toimub midagi olulist – näiteks võib alata tegelase animeerimine, kaamera nurk muutuda või uks avaneda. Markeritele saab anda nimesid, et neid kiire pilguheidu järel selgemini mõistetavamaks muuta. Neid saab erinevates vormides kasutada paljudes Blenderi akendes. Erinevalt võtmekaadritest asetatakse markerid alati täisnumbrilistesse kaadritesse – näiteks ei saa sa markerit panna “kaadrisse 2.5”.

Markereid saab luua ja muuta kõigis järgnevates redaktorites (ja kõigis nende erinevates režiimides):

- [Graafiku redaktor](#).
- [Tegevuste redaktor](#)
- [Tööleht](#)
- [Mittelineaarne \(NLA\) redaktor](#)
- [Videomonteerija](#)
- [Ajatelje aken](#)

Mõnes nendest akendest loodud marker ilmub ka teistes markereid toetavates akendes, muuhulgas:

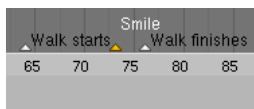
- [3D-vaate aken](#).

Poosimarkerid (*Pose markers*)

On veel üks tüüp markereid, mida kutsutakse “poosimarkeriteks” ning mis on mõeldud spetsiaalselt skelettide ja tegevusredaktori (*Action Editor*) jaoks. Need on seotud pooside teekidega ning neist räägitakse täpsemalt [siin](#).

Visualiseerimine

Tavapärane



Markerid: väikesed, aga kasulikud.

Enamik aknatüüpe kuvab markereid samal moel: alaservas olevad väikesed kolmnurgad, mis on valitutena kollased ja mittevalitutena valged.

Kui neil on nimi, siis näidatakse seda markeri valimisel valges kirjas markerist paremal pool. Vaata pilti (*Markerid: väikesed, aga kasulikud*).

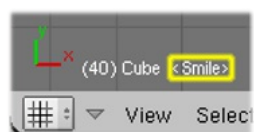
Monteerija (*Sequencer*)



Markerid monteerijas

Videomonteerija (*Video Sequence Editor*) lisab igale markerile lihtsalt vertikaalse punktiirjoone (mittevalitud markeritel hall, valitud markeritel valge).

3D-vaade



Marker 3D-vaates.

3D-vaate aken ei luba sul markereid lisada/muuta/kustutada. See näitab sulle lihtsalt vasakus alumises nurgas aktiivse objekti nime lähedal markeri nime märkide <> vahel, kui oled markeriga märgitud kaadris (vaata pilti *Marker 3D-vaates*).

Markerite loomine ja muutmine

Kahjuks ei ole markerite muutmiseks üldiseid kiirvalikuid või menüüd, mis oleks kõigis neid toetavates akendes ühine... Seega oleme pannud iga nendega seotud tegevuse juures olevasse viitekastis kõige tavapärasema kiirvaliku ja menüü sissekande ning sulgudesse kõik teadaolevad erandid.

Markerite loomine

Mode: Kõik režiimid

Hotkey: M (CtrlAltM videomonteerijas)

Menu: Marker » Add Marker (Frame » Add Marker ajateljel)

Kõige lihtsam moodus markerite lisamiseks on minna kaadrisse, kuhu neid soovid panna, ja vajutada klahvi M (või videomonteerijas



kombinatsiooni CtrlAltM).



Alternatiivina võid kasutada kombinatsiooni AltA (või ajatelje akna “mahamängimise” nuppu), et animatsiooni mängima panna, ning vajutada õigetes kohtades klahvi M (või videomonteerijas kombinatsiooni CtrlAltM). See võib olla eriti kasulik muusika taktide märkimiseks.

Markerite valimine

Mode: Kõik režiimid

Hotkey: RMB , ⇧ Shift RMB , A/CtrlA, B/CtrlB

Vajuta markeri kolmnurga peale klahviga RMB , et seda valida. Kasuta kombinatsiooni ⇧ Shift RMB , et valida mitut markerit või nende valikut tühistada.

IPO-kõverate redaktoris, tegevuse redaktoris, mittelineaarses (NLA) redaktoris ja videomonteerijas võid ka kõigi markerite valimiseks/valiku tühistamiseks kasutada klahvi CtrlA ning neid piirdkastiga kombinatsiooni CtrlB abil valida (LMB  valib ja RMB  tühistab valiku, nagu tavaliselt). Vastavad võimalused on olemas ka nende akende valikumenüüs *Select*.

Ajateljes ja audio aknas saad sa kõik markereid valida/valikut tühistada klahviga A ja piirdkastiga valida klahviga B.

Markerite nimetamine

Mode: Kõik režiimid

Hotkey: CtrlM

Menu: Marker » (Re)Name Marker (ajateljes menüü Frame » Name Marker)



Kui sul on stseenis kümneid markereid, mis tähistavad erinevaid kohti ajas, pole sul nendest eriti palju abi, kui sa ei tea, mida need tähendavad. Markerile saad nime anda selle valides, vajutades CtrlM, trükkides nime ja vajutades seejärel nuppu OK.

Markerite liigutamine

Mode: Kõik režiimid

Hotkey: CtrlG (ajateljes või audio aknas G)

Menu: Marker » Grab/Move Marker (ajateljes menüü Frame » Grab/Move Marker)

Kui oled ühe või mitu markerit valinud, kasuta kombinatsiooni CtrlG (ajateljes või audio aknas klahvi G), et neid liigutada, ning kinnita liigutamine klahvidega LMB  või ↵ Enter (klahvide RMB  või Esc abil saad liigutamise tühistada, nagu tavaliselt).

Vaikimisi liiguvad markerid ühekaadriste sammude haaval, kuid hoides all klahvi Ctrl, liiguvad nad ühele sekundile vastavate sammude haaval – kui su steeni kaadrisagedus on **25 fps**, liiguvad markerid kahekümne viie kaadri pikkuste sammudega.

Markerite duplitseerimine

Mode: Kõik režiimid

Hotkey: Ctrl⇧ ShiftD (ajateljes või audio aknas ⇧ ShiftD)

Menu: Marker » Duplicate Marker (ajateljes menüü Frame » Duplicate Marker)

Valitud markereid saad duplitseerida kombinatsiooniga Ctrl⇧ ShiftD (ajateljes või audio aknas kombinatsiooniga ⇧ ShiftD). Pärast duplitseerimist aktiveeritakse nihutamisrežiim, et saaksid neid panna, kuhu (või õigemini mis aega) soovid.


Pane tähele, et erinevalt enamikust muudest duplitseerimistest Blenderis, ei muudeta duplitseeritud markerite nimesid üldse (neile ei lisata numbrilist lõppu nagu “.001”...).

Markerite kustutamine

Mode: Kõik režiimid

Hotkey: ⇧ ShiftX (ajateljes või audio aknas X)

Menu: Marker » Delete Marker (ajateljes menüü Frame » Delete Marker)

Valitud markeri(te) kustutamiseks vajuta lihtsalt ⇧ ShiftX (ajateljes või audio aknas X) ning kinnita hüpikeates kustutamine klahviga LMB .

Võtmekaadrid

Võtmekaadrid on animatsiooni alus. Võtmekaadrid defineerivad parameetrite väärtused määratud kaadris.

Ilmne on, et miski on "animeeritud" siis, kui ta muutub ajas. Blenderis tähendab objekti animeerimine selle parameetrite, näiteks tema asukoha X-teljel või materjali põhivärvi punase kanali muutmist mingi ajaperioodi jooksul.

Nagu mainitud, on Blenderi funtamentaalne ajaühik "kaader", mis tavaliselt kestab sekundi murdosa, sõltuvalt stseeni "kaadrisagedusest".

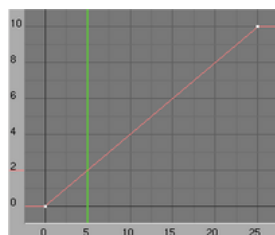
Animatsioon koosneb üksteisega liituvatest muutustest, mis toimuvad mitme kaadri jooksul. Reeglina EI MÄÄRATA neid muutusi *kaaderhaaval*, sest:

- selle tegemine võtaks terve igaviku aega!
- parameetri sujuva muutumise saavutamine oleks väga raske (kui sa just ei arvuta välja muutuste matemaatilisi väärtusi ega kirjuta neid täpselt igasse kaadrisse – mis oleks jabur).

Seetõttu on tehakse peaaegu kõik otsesed animatsioonid **interpolatsiooni** abil.

Selle idee on lihtne: sa defineerid mõned üksteisest mitme kaadri kaugusel asetsevad "juhtpunktid", mida nimetatakse "**võtmekaadriteks**" (*keyframes*).

Blender arvutab (interpoleerib) parameetrite väärtused nende juhtkaadrite vahel ja määrab vastavas kaadris animeeritavale kanalile. Seega on animaatoril oluliselt vähem tööd. Neid [F-kõvera](#)id saab muuta Doc:ET/2.6/Manual/Animation/Graph Editor[graafikuredaktoris].



Interpolatsiooni näide

Näiteks kui sul on:

- juhtpunkti väärtus kaadris **0** on **0**,
- järgmine väärtus kaadris **10** on **25**,
- lineaarne interpolatsioon,

siis kaadris **5** saame me väärtuseks **2**.

Sama kehtib kõikide vahepealsete kaadrite kohta: kõigest kahe punkti määramisega saad sa **25 kaadri** jooksul sujuva tõusu "0"-ist "10"-ni. Kui sa aga tahad näiteks, et parameetri väärtus kaadris **15** oleks **9**, tuleb sul lisada täiendav juhtpunkt (ehk võtmekaader)...

Võtmekaadrite tekitamine


3D-vaates

Võtmekaadreid luuakse kiirklahviga I. Avaneb menüü järgnevate valikutega:

- Location (asukoht)
- Rotation (pööre)
- Scale (skaala)
- LocRot (asukoht + pööre)
- LocScale (asukoht + skaala)
- LocRotScale (asukoht + pööre + skaala)
- RotScale (pööre + skaala)
- Visual Location (nähtav asukoht)
- Visual Rotation (nähtav pööre)
- Visual Scale (nähtav skaala)
- Delta Location (asukoha muut)
- Delta Rotation (pöörde muut)
- Delta Scale (skaala muut)

Omaduste paneelil

Võtmekaadreid saab tekitada peaaegu igale muudetavale omadusele. Sa saad seda teha ühel järgmistest viisidest:

- hoides hiirekurssorit omaduse välja kohal, kasuta kiirklahvi I. Väli muutub kollaseks, näidates, et juhtkaader on määratud.
- Vajuta vektoromaduse väljal RMB  ja vali Insert Keyframes (lisa võtmekaadrid) või Insert Single Keyframe (lisa üks võtmekaader).

Animatsiooniredaktorites

Võtmekaadreid saab lisada juba olemasolevate võtmekaadritega kanalitele animatsiooniredaktorites. Seda kirjeldatakse vastavatel lehekülgedel:

- [Ajatelg](#)
- [Graafikuredaktor](#)

- [Tööleht](#)

Võtmekaadrite muutmine

Võtmekaadrite muutmiseks on mitmeid mooduseid, mida kirjeldatakse ülalpool loetletud teistel lehekülgedel.

Animatsiooniredaktorid

Animeerimisega on omajagu tööd. Enamus sellest on rutiinne. Seetõttu on aastate jooksul välja töötatud mitmeid erinevaid tehnikaid animaatori töö kiirendamiseks ja lihtsustamiseks.

Kõikide nende võimaluste haldamiseks on Blenderis mitu animatsiooniredaktorit:

[Ajatelg](#)

Ajatelg ei võimalda animatsiooni andmete muutmist, ent näitab kõik stseeni võtmekaadreid. Ajaskaalal saab muuta [animatsioonimarkereid](#).

[Graafikuredaktor](#)

Madalaima taseme redaktor, kus saab kõvera kuju abil määrata iga animeeritava omaduse/väärtuse/kanali muutumise.

[Tööleht](#)

Töölehel (endine *Action Editor*) on neli töörežiimi, mis on väga sarnased, kuid töötavad erinevate animatsiooniandmete tüüpidega:

[Töölehe](#) režiim

Üldine võtmekaadrite redaktor. Näitab kõigi stseeniobjektide kõiki võtmetega kanaleid.

[Liikumisedaktor](#) režiim

Sarnane töölehe režiimile, kuid näitab valitud objekti juhtkaadreid ja on mõeldud spetsiaalselt tegevuste loomiseks.

[Võtmevormide redaktor](#) režiim

Näitab valitud objektide võtmevormidega animatsioone.

[Rasvapliiatsi](#) režiim

Näitab eelnevalt rasvapliiatsiga tehtud animatsiooni mustandit.

[Mittelineaarne \(NLA\) redaktor](#)

Selle kõrgtaseme redaktori idee on laenatud mittelineaarsest videomontaažist: iga element tähistab tegevust, mida sa saad oma suva järgi liigutada, duplitseerida, kahandada/kasvatada (sh kiirendada/aeglustada) jne. Kui mõelda võtmetest nagu võre külgedest, siis 'NLA tegevus' on seda võret ümbritsev töötlejaga objekt.



Oluline on aru saada, kuidas animatsiooni kulg kajastub animatsiooniredaktorites:

- võtmekaadrid moodustavad animatsioonikõverad,
- mida kasutatakse **tegevuste** loomiseks,
- mida kasutab **NLA**.

F-kõveratega töötamine (*F-Curves*)

Kui sa oled millegi jaoks loonud võtmekaadrid, saad nendele vastavaid kõveraid muuta. Blenderi versioonis 2.5 on IPO-kõverad (interpoleeritud kõverad) asendatud F-kõveratega, kuid nende kõverate muutmise on sisuliselt siiski sama.

Interpoleerimise põhimõte

Kui midagi "animeeritakse", siis see muutub ajas. Blenderis tähendab objekti animeerimine mõne selle omaduse muutmist teatud aja jooksul (näiteks X-asukoht, selle materjali hajusvärvi punase kanali väärtus jne).

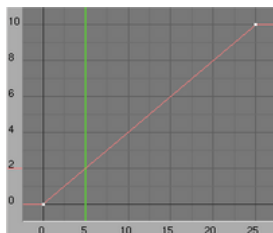
Nagu mainitud, on Blenderi fundamentaalne ajaühik "kaader", mis tavaliselt kestab sekundi murdosa, sõltuvalt stseeni "kaadrisagedusest" (*frame rate*).

Animatsioon koosneb üksteisega liituvatest muutustest, mis toimuvad mitme kaadri jooksul. Reeglina EI MÄÄRATA neid muutusi *kaaderhaaval*, sest:

- selle tegemine võtaks terve igaviku aega!
- parameetri sujuva muutumise saavutamine oleks väga raske (kui sa just ei arvuta välja muutuste matemaatilisi väärtusi ega kirjuta neid täpselt igasse kaadrisse – mis oleks jabur).

Seetõttu on tehakse peaaegu kõik otsesed animatsioonid **interpolatsiooni** abil.

Selle idee on lihtne: sa defineerid mõned üksteisest mitme kaadri kaugusel asetsevad võtmekaadrid (*keyframes*). Blender arvutab (interpoleerib) parameetrite väärtused nende juhtkaadrite vahel ja paneb need ise paika. Seega on animaatoril oluliselt vähem tööd.



Interpolatsiooni näide

Näiteks kui sul:

- juhtpunkti väärtus kaadris **0** on **0**,
- järgmine väärtus kaadris **10** on **25**,
- lineaarne interpolatsioon,

siis kaadris **5** saame me väärtuseks **2**.

Sama kehtib kõikide vahepealsete kaadrite kohta: kõigest kahe punkti määramisega saad sa **25 kaadri** jooksul sujuva tõusu "0"-ist "10"-ni. Kui sa aga tahad näiteks, et parameetri väärtus kaadris **15** oleks **9**, tuleb sul lisada täiendav juhtpunkt (ehk võtmekaader)...

Interpolatsiooni tüübid

Hoolimata nimetusest "kõver", võib see olla sõltuvalt valitud interpolatsiooni tüübist lineaarne segmentide komplekt või koosneda isegi mittepidevatest väärtustest (moodustades "trepikujulise kõvera").

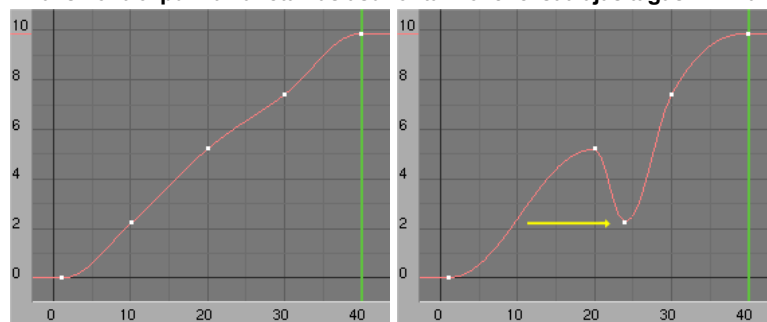
Aja suund

Kuigi F-kõverad on väga sarnased [Bezier' kõveratele](#), on mõned olulised erinevused.

Mõistetavatel põhjustel **ei saa kõveral kujutatud omadusel olla samal ajal mitut erinevat väärtust**, seega:

- kui sa liigutad mõnda kontrollpunkti ette oole mõnest kontrollpunktist, mis oli enne liigutatavast punktist eespool, vahetavad need kaks punkti kõveral oma asukohad, et kõver ei liiguks ajas tagasi
- eelnevast põhjusest tulenevalt on suletud IPO-kõver võimatu

Kaks kontrollpunkti vahetamas asukohta: kõver ei saa ajas tagasi minna!





Enne teise võtmekaadri liigutamist




Pärast teise võtmekaadri liigutamist

Kõvera valimine

Kui sa kanalile võtmekaadreid lisad, siis näitab graafikuredaktor (*Graph Editor*) sulle neid vaikimisi muutmisrežiimis *Edit*. Nagu 3D-vaateski, saad sa kanali muudetavust sisse ja välja lülitada klahviga \leftrightarrow Tab. Kanalite karbis *Channel Box* toimib luku ikoon sama moodi muudetava ja "mittemuudetava" režiimi vahelise lülitina.

Valimise tööriistad

Kõvera sangasid saad valida neile klahviga LMB  vajutades (või kasutades mitme valimiseks kombinatsiooni ⇧ Shift LMB ). Terve võtmekaadri saad valida keskmise punkti valimisel, kuid on võimalik valida ka ainult üks sang.

Kõiki nähtavaid kõveraid saab valida (või nende valikut tühistada) klahviga A (või menüüga Select » Select/Deselect All) või kasutada klahviga B (või menüüga Select » Border Select) piirdkastiga valimist – valiku loomiseks vajuta klahvi LMB  ja lohista hiirt ning valikust eemaldamiseks kasuta klahvi RMB  (või kombinatsiooni CtrlAlt LMB .

Select All (vali kõik, A)

Invert Selection (pööra valik ümber, CtrlI)

Border Select (piirdkastiga valimine, B)

Border Axis Range (piirdkasti teljed, AltB)

Border (include Handles) (piirdkast koos sangadega, CtrlB)

Columns on Selected Keys (valitud võtmekaadrite tulpad, K)

Column on current Frame (valitud kaadri tulp, CtrlK)

Columns on selected Markers (valitud markerite tulpad, ⇧ ShiftK)

Between Selected Markers (valitud markerite vahel, AltK)

Before Current Frame (enne valitud kaadrit, I)

After Current Frame (pärast valitud kaadrit, J)

Select More (vali rohkem, Ctrl+ NumPad)

Select Less (vali vähem, Ctrl- NumPad)

Select Linked (vali lingitud, L)

Kõverate muutmine

Teisendused

Sa saad *valitud* kõveraid (need, mis on nähtavad ja mille kontrollpunktid/võtmekaadrid on valged) nihutada (G), pöörata (R) ja nende mõõtkava muuta (S) – neid tegevusi saab läbi viia ka kõvera teisendamise alammenüü Curve » Transform abil.

Lisaks võid sa nihutamise või mõõtkava muutmise puhul piirata teisenduse X- (aeg) või Y-teljega (väärtus), vajutades nagu tavaliselt teisenduse ajal klahve X või Y.

Kasutada saad ka klassikalist “teisenduse nakkumise” võimalust: klahvi Ctrl all hoidmine teisendamise ajal nakkub ühe kaadri/ühe väärtuse sammu haaval ning klahvi ⇧ Shift all hoidmine aeglustab liikumist ja suurendab seega täpsust. Pane samas tähele, et kombinatsioon ⇧ ShiftCtrl aeglustab ainult liikumist ilma nakkumise sammudeta.

Kopeerimine

Saad kopeerida ühe või enam võtmekaadrit puhvrisse ning kleepida nad seejärel *samadele* kõveratele. Seda tehakse kahe päises oleva nupu abil:

- “Kopeerimise” (CtrlC) nupp (nool alla) kopeerib *valitud ja nähtavad* kõverad puhvrisse.
- “Kleepimise” (CtrlV) nupp (nool üles) kleepib puhvri sisu valitud kanali *nähtavatesse* kõveratesse.

Pane tähele, et see tööriist ei tööta “võtmekaadrite” režiimis (kuigi on seal olemas). Ja muutmisrežiimis kopeeritakse/kleepitakse ainult valitud punktid (mitte terve kõver).

Sa saad võtmekaadreid ka võtme menüü Key abil **duplitseerida** (Duplicate).

Kusutamine

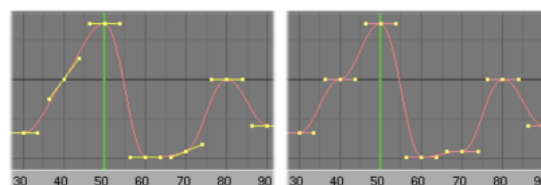
Võtmekaadri kustutamiseks vali see (“vakimisi” režiimis) ja vajuta klahvi X (või kasuta menüüd Curve » Delete).

Nakkumine

Sul on ka mõned “nakkumise” valikud (⇧ ShiftS või Curve » Snap):

- Horizontal (horisontaalne) – See on kasulik ainult vaikimisi F- kõverate puhul: pöörab valitud kõverate kontrollpunkte nii, et need oleksid horisontaalsed.

“Horisontaalse” nakkumise näide.

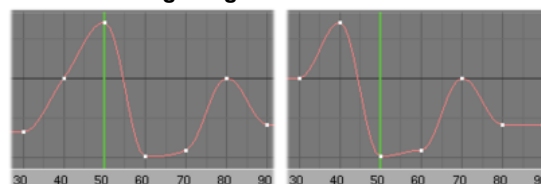


Enne horisontaalset
(Horizontal) nakkumist.

Pärast horisontaalset
(Horizontal) nakkumist.

- To Next (järgmisega) – See määrab kõigile valitud kõverate kontrollpunktidele (võtmekaadritele) sama kõvera järgmise kontrollpunkti väärtuse (tekitaes teatud sorti kõvera “nihkumise”).

“Järgmisega” nakkumise näide.



Enne järgmisega (To Next)

Pärast järgmisega (To Next)

nakkumist.

nakkumist.

- To Frame (kaadriga) – See liigutab valitud kõverate kõiki võtmekaadreid horisontaalselt (ajas) lähima kaadri (nt võtmekaader, mis on loodud kaadris **23.2**, liigub kaadrisse **23**).
- To Current Frame (praeguse kaadriga) – Sellel valikul ei ole “vaike”režiimis mingit mõju.

Peegelda (*Mirror*)

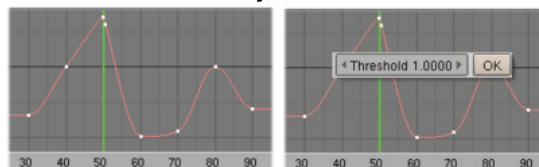
Sa saad valitud kõveraid peegeldada (⇧ ShiftM või Curve » Mirror) kas üle praeguse kaadri (Over Current Frame, st roheline kursori), üle vertikaaltelje (Over Vertical Axis, st kaadri **0**) või üle horisontaaltelje (Over Horizontal Axis, st väärtus **0.0**).

Puhastamine ja silumine

Sa saad valitud IPO-kõveraid puhastada (st eemaldada kontrollpunkte, mis on nii ajalisel kui ka väärtuse poolest üksteisele väga lähedal). Vajuta klavi O (või kasuta menüüd Curve » Clean Keyframes), määra hüpikdialoogis sobiv lävi (*Threshold*) ja vajuta nuppu OK.

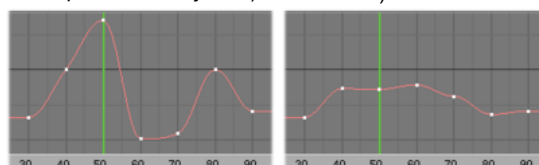
On olemas ka võimalus valitud kõverate silumiseks (AltO või Curve » Smooth Keys), kuid pea meeles, et selle algoritm näib ka ilma ühegi säteteta jagavat iga võtmekaadri vahelise kauguse kahega ja leidvat kõvera keskmise lineaarse väärtuse ning seetõttu on silumine suhteliselt võimas! Pane ka tähele, et see tööriist ei näi kunagi muutvat esimest ja viimast võtmekaadrit.

Puhastamise ja silumise näited.



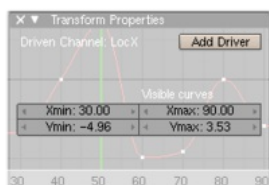
Ebapuhas kõver (pane tähele kahte üksteise lähedal olevat punkti kursori juures).

Selle puhastamine (pane tähele suhteliselt kõrget läve *Threshold*).



Kui kõver on nüüd puhas...

...saame seda siluda!



Kõverate teisendussätete paneel Transform Properties “vaike”režiimis.

Teisendussätete paneel Transform Properties

Jah, see toimib nagu 3D-vaates (ning seda näidatakse/peidetakse samuti klaviviga N või menüüvalikuga View » Channel Properties)! “Vaike”režiimis ei ole see eriti kasulik, kuna see näitab (ja saab muuta) ainult korraga kõigi nähtavate kõverate X/Y miinimumi/maksimumi ning seega lubab kõiki neid kõveraid korraga suurendada/vähendada.

Sample Keyframes (sämpli võtmekaadreid, ⇧ ShiftO)

Sämplib valitud võtmekaadreid ning asendab interpolateeritud väärtused iga kaadri jaoks eraldi võtmekaadriga.

Bake Curves (valmista kõverad, AltC)

Kõvera valmistamine (*baking*) asendab selle sämplitud punktidega ning keelab kõvera muutmise.

Interpolatsioon ja ekstrapolatsioon

F-kõveratel on veel kolm lisasätet, mis määravad punktide vahelise interpolatsiooni, kõvera pikendamise käitumise ja sangade tüübi.

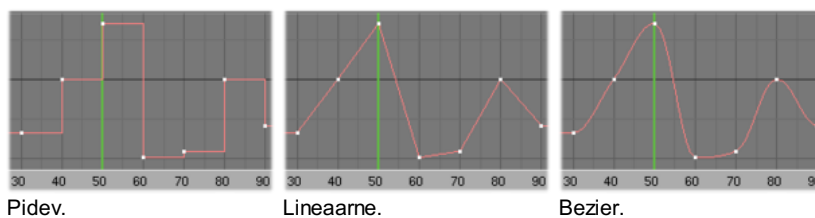
Interpolatsioon (*Interpolation*)

Sul on kolm valikut (T või Curve » Interpolation Mode):

- Constant (pidev) – Interpolatsiooni ei toimu üldse. Kõveral on pidevalt viimase võtmekaadri väärtus, mis tekitab astmelise (treppis) “kõvera”. See on väga harva kasulik.
- Linear (lineaarne) – See lihtne interpolatsioon loob kahe naabruses oleva võtmekaadri vahele sirge lüli, mille tulemuseks on katkendjoon. See võib olla väga kasulik, kui kasutada ainult kahte võtmekaadrit ja pikendamise ehk ekstrapolatsiooni režiimi Extrapolation, mille tulemuseks on lõpmatu sirge joon (st lineaarne kõver).
- Bezier – Kõige võimsam ja kasulikum interpolatsioon, mis on ka vaikimisi määratud. Selle tulemuseks on kenad sujuvad kõverad ja seega sujuvad animatsioonid!

Pea meeles, et mõned IPO-kõverad saavad kasutada ainult mittepidevaid väärtusi, mistõttu neid näidatakse hoolimata valikust alati, justkui nad oleksid pideva interpolatsiooniga.

Kolm kõverate interpolatsiooni.



Ekstrapolatsioon (*Extrapolation*)

Ekstrapolatsioon määrab kõvera käitumise pärast esimest ja viimast võtmekaadrit.

Ekstrapolatsiooni režiime on kaks (⇧ ShiftE või Channel » Extrapolation Mode):

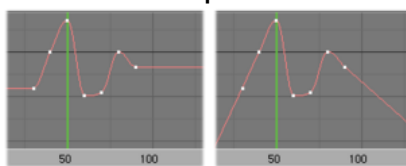
Constant (pidev)

Vaikimisi valitud. Enne esimest ja pärast viimast võtmekaadrit on kõveratel pidev väärtus (nende samade võtmekaadrite oma).

Linear (lineaare)

Kõvera lõpud on sirged jooned (lineaarsed), mis on määratud nende esimese ja viimase kahe kontrollpunkti poolt.

Kaks IPO-kõverate pikendamise režiimi.



Ekstrapolatsiooni lisatööriistad asuvad [F-kõvera töötlejate all](#)

Sanga tüübid

Bezier' interpolatsiooniga kõverate puhul on veel üks suhteliselt kasulik võimalus. Saad määrata, millist tüüpi sangasid kõvera punktide puhul kasutatakse V

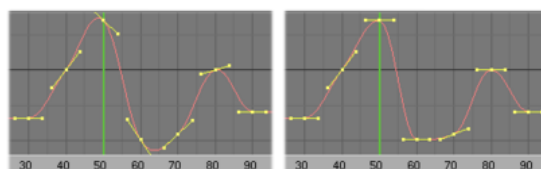
Automatic (automaatne)

Vector (vektor)

Aligned (joondatud)

Free (vaba)

Auto Clamped (automaatselt piiratud)



Juhtijad (*Drivers*)

Juhtijaid kasutatakse ühe omaduse animeerimiseks sõltuvalt teise omaduse väärtusest. See tähendab, et nende animatsiooni väärtust ei määra mitte kaadri number, vaid valitud kanal. Juhtijad saavad oma efekti aluse võtta üksikust omadusest, pöörde erinevusest või skriptitud Pythoni avaldisest, mida saab otse kasutajaliideses muuta.


Sa võid näiteks: kasutada mõne juhtija X-asukohta, et muuta mõne materjali värve (RGB-kõveraid); kasutada juhtija pööret, et määrata mõne objekti mõõtkava; kasutada juhtija mõõtkava, et määrata mõne võre/kõvera/jne kuju (läbi võtmevormide (*shape keys*)); kasutada Pythoni funktsiooni, et määrata mõne piiraja mõju; ja palju-palju muudki veel...

Üks peamiseid juhtijate kasutusvaldkondi on tegelaste animeerimine: võid näiteks lisada juhtija objekte (*object drivers*) näo suhtelistele võtmevormidele (*relative shape keys*). Seejärel muudad sa oma tegelase ilmet lihtsalt neid juhtija objekte liigutades (mis on loomulikult mitterenderduvad!). See on natuke sarnane mittelineaarsetele animatsioonile (NLA), kuid kasutajasõbralikum! Vaata [juhitavaid võtmevorme \(Driven Shape Keys\)](#).

Kuigi nende liides ja toimimine on sarnased F-kõvera (*F-curve*) omadele, on juhtijad väljaspool tegevusi (*Actions*), olles viimastega animatsioonisüsteemi hierarhias võrdsel tasemel. Juhtijaid arvutatakse pärast tegevusi, seega võivad juhtijate efektid f-kõveraga tegevuste sees tehtud animatsioonid üle kirjutada.

Laotus (*Mapping*)

Juhtivat kõverat ei määrata ajaliselt (jah, me kordame, kuid selle mõistmine on *väga* oluline). Need on hoopis juhtija omaduse muutused, mis määravad, millises kohas kõvera X-teljel selle väärtusi tuleks arvestada (st kõver laotab juhtija omaduse väärtuse ümber kõvera omaduse väärtuseks).

Vaikimisi toimub laotamine sisendväärtuselt väljundväärtusele üks-ühele ning selle määrab f-kõvera loomise töötleja (*Generator F-Curve Modifier*). Saad luua isetehtud laotuse, kustutades selle töötleja ning sisestades kombinatsiooni Ctrl + LMB  abil kõvera graafiku allasse võtmekaadreid.

Juhtijate loomine

Juhtija loomiseks leia omaduste paneelis omadus, mida soovid juhtida. See saab selleks omaduseks, mida teise omaduse poolt **juhitakse**.

Vajuta klahviga RMB  kanali peal ning vali üks järgnevatest võimalustest:

Add Drivers (lisa juhtijad)

See loob valitud kanaliga seotud kanalitele juhtijad. Näiteks lisab see X, Y ja Z nihutamisele juhtijad.

Add Single Driver (lisa üks juhtija)

See lisab ainult valitud kanalile ühe juhtija.

Juhtijate muutmine

Pärast juhtija loomist pead sa määrama, kuidas seda kanalit juhitakse ja mille poolt juhtimine toimub.

Juhtijate redaktor asub graafiku redaktoris (*Graph Editor*). Saad minna f-kõvera redaktori (*F-Curve Editor*) vaatest juhtijate (*Driver*) vaatesse, kasutades päises olevat menüüd.

Juhtijate redaktor näitab kõiki avatud stseenis olevaid juhtijaid. Nende nimekiri asab vasakul küljel olevas kanalite paneelis *Channel*.

Kui sul on stseenis rohkem kui üks juhtija, saad siit lülitada välja ja sisse nende kuvamist graafiku alas. Samuti pead sa valima mõne kindla kanali, et seda muutma asuda.

Juhtija omadused

Juhtija omadused asuvad paremal küljel olevas omaduste paneelis *Properties*.

Update Dependencies (värskenda sõltuvusi)

Värskendab juhtija ühendusi.

Juhtijate tüübid

On olemas mitu erinevat juhtijate tüüpi, mis määravad selle, kuidas juhtija sisendi väärtust arvutatakse:

Maximum Value (maksimaalne väärtus)

Võtab kasutaja muutujatest maksimaalse väärtuse.

Minimum Value (minimaalne väärtus)

Võtab kasutaja muutujatest minimaalse väärtuse.

Sum Values (väärtuste summa)

Võtab kasutaja muutujate summa.

Averaged Values (keskmine väärtus)

Võtab kasutaja muutujate keskmise.

Scripted Expression (skriptitud avaldis)

Arvutab väärtuse tulenevalt lihtsast Pythoni avaldisest, mis omakorda võib kasutada kasutaja muutujaid. (loe altpoolt)

Muutujad (*Variables*)

Kasutaja muutujad loovad juhtija sisendväärtuse. Kui sa ei kasuta skriptitud avaldist, peab sul olema vähemalt üks muutuja. Kasutades ühte muutujat, loovad kõik ülalpool toodud juhtijate tüübid (v.a skriptitud avaldis) sama väärtuse.

Üksikomadus (*Single Property*)

Kasuta üksikomaduse tüüpi *Single Property*, et leida ükskõik millise antud stseeni koostisosa suvaline unikaalne andmeblokk. Selleks võib olla ükskõik mis: alates valgusti intensiivsusest kuni mõne tekstuuri värvini.

Teisenduskanal (*Transform Channel*)

See tüüp lubab sul muuta kõveraid läbi mõne objekti teisendusomaduste - seega saab üks juhtija kontrollida korraga kuni üheksat kõverat (juhtija asukoha, pöörde ja mõõtkava kolm komponenti).

Juhtija objekti loomiseks:

- Vali kanal, mida soovid juhtida.
- Sisesta ilmuvasse objekti välja OB soovitud juhtija objekti nimi.
- Kui su juhtija ei ole skelett, on sul juhtija tüübi rippmenüüs Driver type ainult objekti sissekanne Object.
- Viimaks vali teisest rippmenüüst omadus, millega soovid oma kõverat juhtida (Loc X, Scale Z jne).

Nüüd ei määra su kõverat enam aeg, vaid sinu poolt valitud omaduse väärtus. Väärtused on Blenderi ühikutes.

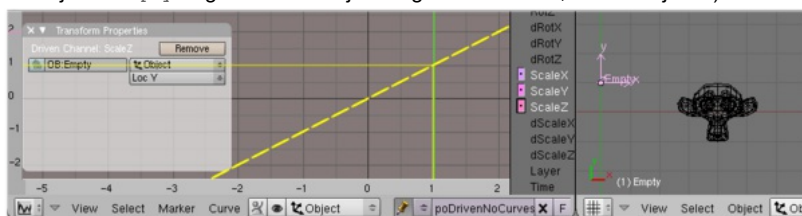
Pane tähele, et juhtija omadusi arvutatakse kohalikus ruumis (st tema ülemobjekti teisendused ei lähe arvesse).

Näide

Selles näites määrame me tuntud pärdikupea (*Suzanne*) mõõtkava tühiobjektist *Empty* juhtija Y asukohaga. Seega lisame me nupuga Add Driver objekti *Suzanne* mõõtkava kanalitele ScaleX, ScaleY ja ScaleZ juhtija (kui vastavat kõverat veel ei eksisteeri, siis luuakse see automaatselt). Pane tähele, et antud hetkel **kõverat olemas ei ole** ning seega lisab Blender üks-ühele laotuse, justkui eksisteeriks ühtlaselt kasvav lineaarne kõver (mida kujutatakse allolevatel pildidel kollaste punktiirjoontena). Need pildid näitavad ka seda, et saad kasutada sama juhtija omadust (antud juhul Y asukohta) erinevate juhtijate jaoks...

Juhtija objektid.

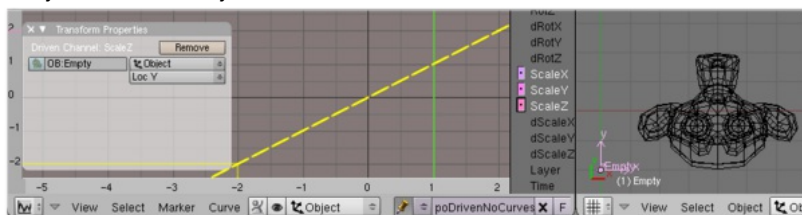
Pane tähele, et kõveraredaktori aken (Curve Editor) on lukustatud objekti *Suzanne* f-kõverale (mistõttu saame haarata kinni tühiobjektist *Empty* ning näha samal ajal ikkagi neid kõveraid, mida see juhib!).



Tühi juhtija asukohas $Y = 1.0$, mis määrab Suzanne'i mõõtkavaks $1.0 \dots$



Juhtija asukoht $Y = 2.0$, juhitava mõõtkava $2.0 \dots$



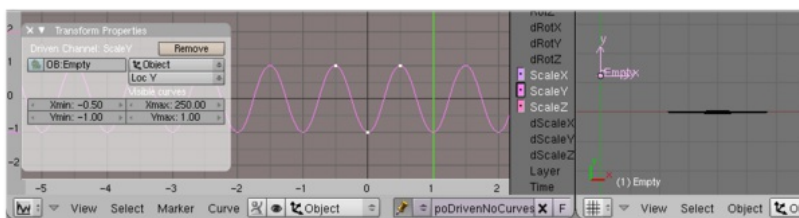
Ja juhtija asukohas $Y = -2.0$, juhitava mõõtkava... Eks sa vist said aru!

Nüüd saame juhtijate juures kasutada kõveraid, mille tulemuseks on keerukam käitumine kui lihtne üks-ühele laotus. Allpool on meil sama stseen mis üleval, kuid kanalis ScaleY on meil pseudosinusoidne kõver, mis võngub ühe kaadri jooksul väärtuste 1.0 ja -1.0 vahel ning mis on loodud kolme kontrollpunkti ja tsükilise ekstrapolatsiooniga (Cyclic *extrapolation*, E). See tähendab, et kuigi objekti *Suzanne* mõõtkava X ja Z komponente määrab ikka veel objekti *Empty* Y asukoht, siis mõõtkava Y komponent varieerub nüüd tsükiliselt väärtuste 1.0 ja -1.0 vahel. Nagu illustreerivad ka järgnevad pildid.

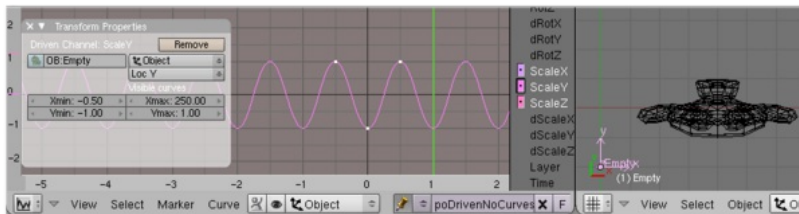
Kõveraga juhtija objekt.



Tühi juhtija asukohas $Y = 1.0$ tekitab Suzanne'i mõõtkavaks $(1.0, 1.0, 1.0) \dots$



Tühi juhtija asukohas $Y = 1.25$ tekitab juhitava mõõtkavaks $(1.25, 0.0, 1.25)$...



Ja tühi juhtija asukohas $Y = -2.0$ tekitab juhitava mõõtkavaks $(-2.0, 1.0, -2.0)$!

Pöörde erinevus (Rotational Difference)

Pöörde erinevused töötavad ainult luude puhul, kuid need toimivad peaaegu samuti nagu objektid omad.

Sa määrad juhtija luu, sisestades objekti välja OB mõne skeleti nime ning valides seejärel juhtija tüübi rippmenüüst Driver type poosi valiku Pose. Siis sisesta luu nimi ilmuvasse luu välja BO ja vali selline omadus, mida soovid kasutada.

Pane tähele järgnevaid erisusi:

- Teisendusi arvutatakse luu enda ruumis (st ülemluu ning terve objekti teisendused ei avalda mitte mingisugust mõju).
- Kasutades juhtija omadusena pööret, on sul üks piirang: kasutatava luu pööre konverteeritakse vahemikku $[-180^\circ, 180^\circ]$ (mis tähendab, et kui sa näiteks pöörad luud üle 180° , hüppab juhitav objekt tagasi X-kõvera asukohta -180° ...).
- Saad kasutada ühte juhtija lisaomadust: pöörde erinevus (Rotation Difference) laseb sisestada mõne teise luu nime ning kasutab juhtijana nende kahe luu vahelist nurka.

Kaugus (Distance)

Kauguse režiim võtab lihtsalt kahe objekti või luu vahelise kauguse

Avaldisjuhtijad (Expression Drivers)

Pythoni avaldisjuhtijad (ehk lühidalt ka *Pydrivers*) lubavad sul mõne teise objekti omaduse asemel (nagu tavalised juhtijad) kasutada kanali sisendina üherealisi [Pythoni](#) avaldise. Avaldis tähendab programmeerimisel mingit sümbolite kombinatsiooni, millega saab leida mingi kindla väärtuse.



Nende avaldisjuhtijatega ei määra sa otseselt kõvera väärtust, vaid seda (nagu teistegi juhtija tüüpide puhul), kust kohast *kõvera X-teljel* tuleks väärtus leida

Avaldisjuhtija loomiseks muuda juhtija tüüp skriptitud avaldiseks (*Scripted Expression*) - seejärel näed tavalist tekstivälja, millesse saad trükkida oma Pythoni avaldise.

Need juhtijad pakuvad mitmeid huvitavaid võimalusi: saad kasutada objektide animatsiooni juhtimiseks matemaatilisi funktsioone ja üldisemat programmeerimist. Ainus piirang on see, et avaldisjuhtija avaldis *ise* peab andma tulemusena mingi tõelise numbri ning mitte näiteks tekstistringi või millegi muu.

Näiteks võid kasutada siinuseid või koosinusi, et objekte ümber mingi punkti võnkuma panna, või igas animatsiooni kaadris juhusliku väärtust, et mõnda materjali omadust muuta (nagu hajususe RGBd, alfa jne).

Pane tähele, et avaldisjuhtija väärtuse leidmine toimib samuti nagu sisesehitatud funktsioon `eval()`, kuid selle piiranguga, et avaldisjuhtija avaldis tohib anda tulemuseks ainult päris numbreid (igat tüüpi number, mida saab automaatselt teisendada ujukomaga arvaks).

Valiidsed avaldised

Oleme sulle algtõdedest juba rääkinud: on tekstiväli, kuhu saad trükkida Pythoni avaldise. Siin on mõned valiidsed avaldised:

- Iga tõeline väärtus:
 - `1.0`
- Numbrite ja tehtemärkidega avaldised:
 - `1. 4.5 + 8.9 * 7.0 - (2 / 3.0)`
- Muutujatega avaldised:
 - `1. math.pi * 2 + 5.0`
- Olemasolevad andmed:
 - `1. Blender.Get("curframe") # hetke animatsiooni kaader`

- Natuke matemaatikat:

```
1. math.sin(Blender.Get("curframe")) # hetke kaadri siinus
```

- Animeeritava (juhitava) objekti enda kasutamine läbi konstandi `self`:

```
1. self.LocX * 10
```

Sisseehitatud võimalused ja aliased

Avaldisjuhtijad kasutavad omaenda globaalset sõnastiku, mis on puhverdatud ning mis luuakse uuesti ainult siis, kui mõni avaldis ei tööta. Sellesse sõnastikku on mõned moodulid eelnevalt imporditud, et neid saaks avaldisjuhtija avaldiste poolt kasutada.

Märkus

Hoidmaks kokku trükkimise aega ja avaldise väiksematena, oleme lisanud iga mooduli jaoks aliased: Blenderile saab viidata kas koodiga "Blender" või lihtsalt "b". Allpool on ära toodud kõik moodulid koos kasutatavate aliaastega.

- Kõik sisseehitatud moodulist `builtin` (vaikimisi sisseehitatud moodul `builtin`)
- `Blender`: `blender`, `b`
- `Blender.Noise` (**müra**): `noise`, `n`
- `math` (**matemaatika**): `math`, `m`

Näidisavaldis:

```
1. m.cos(m.pi * b.Get("curframe") / n.random())
```

Aliased on lisatud ka mõnele tihti kasutatavatele andmetele:

- `ob(nimi)` on võrdne objektikoodiga `Blender.Object.Get(nimi)`
- `me(nimi)` on võrdne võrekoodiga `Blender.Mesh.Get(nimi)`
- `ma(nimi)` on võrdne materjali koodiga `Blender.Material.Get(nimi)`

Näidisavaldis:

```
1. ob("Cube").LocX + ob("Lamp").RotZ
```

Blenderi tekst nimega `pydrivers.py`

Lisaks ülaltoodud moodulitele imporditakse ka Blenderi tekstipuhver nimega "`pydrivers.py`", kui see on tekstiredaktorisse laetud:

- `pydrivers`: `pydrivers`, `p`

See võimaldab kasutajatel luua omaenda funktsioone ja lisada omaenda muutujaid ilma üherealiste Pythoni avaldiste piiranguteta. Kui su `pydrivers.py` tekst näeb välja selline:

```
1. myvar = 10.0
2.
3. def myfunction(arg):
4.     # tee siin midagi jube lahedat
5.     return float_val
```

Sa saad kasutada igas avaldises nii muutujat `myvar` kui ka funktsiooni `myfunction`:

```
1. p.myvar * p.myfunction(2) - 1
```

Märkus: kui sa faili `pydrivers.py` teksti täiendad, mine kõveraredaktorisse (Curve Editor) ja vajuta ükskõik millise avaldisjuhtija teksti sisestamise kasti (teisendussätete paneeli Transform Properties), seejärel klõpsa hiirega sellest väljaspoole ning vajuta klahvi ↵ Enter, et sundida moodulit `pydrivers.py` ennast uuesti laadima ja värskendada kõiki avaldisjuhtijaid korraga.

Näidis

Ütleme, et soovime panna pärdiku Suzanne täiuslikus kahe Blenderi ühiku raadiusega ringis mööda XY-tasapinda liikuma. Seda ei ole tavaliste kõveratega just eriti kerge saavutada, sest sa pead looma täiuslikult sinusoidsed kõverad (meeldetuletuseks nendele, kes on matemaatika ära unustanud, et täiusliku ringi saavutamiseks kartesiaanlikul tasapinnal tuleb kasutada valemeid $x = \sin(t)$ ja $y = \cos(t)$...). Seega kasutame Pythoni juhtijaid:

Matemaatika (`math`) funktsioonid `sin()` ja `cos()` kasutavad sisendina *radiaani* väärtust (st täisringi tegemiseks kulub 2π kaadrit) ning väljastavad numbrite vahemiku `[-1.0, 1.0]`. Seega **2 Blenderi ühiku** radiusega ringi saamiseks, mis läbitaks **50** kaadriga (PAL video puhul kaks sekundit), peame kasutama kahte järgnevat Pythoni avaldist:

- X-asukoht `LocX`:

```
1. 2.0 * math.sin(Blender.Get("curframe")*math.pi/25)
```

- Y-asukoht `LocY`:

```
1. 2.0 * math.cos(Blender.Get("curframe")*math.pi/25)
```

Nüüd liigub pärdik Suzanne lõpututes ringides ümber maailma keskpunkti (kui soovid teda panna tiirlema ümber mõne muu punkti, lisa lihtsalt eelnevatele avaldistele nihe või juhtija kõverad `dLoc.....`).

Pane tähele:

- Eelnevas näites ei kasutanud me f-kõverat.
- Täiuslikult ringikujulise liikumise võid saavutada ka, kasutades [liikumistrajektorina NURBS-kõvera ringi](#)...

Lingid

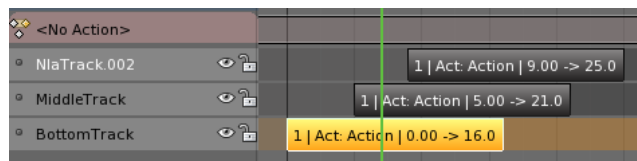
- [Blenderi versiooni 2.49 Pythoni programmeerimisliidese \(API\) juhend](#) ja peatükk [Blenderi Pythoniga laiendamise teemal](#).
- [Python](#) ja selle [dokumentatsioon](#).
- Nendele, kes otsivad funktsioone, mida avaldisjuhtijatega katsetada: <http://functions.wolfram.com/> (algajatel soovitame alustada elementaarsete funktsioonide ja eriti trigonomeetriaga).
- Viimasena täienduste nimekirja sissekannne, kus on nii täiendus (*patch*) kui ka avaldisjuhtijate näidis-.blend-failid: [asub siin](#).

Mittelineaarse animatsiooni (NLA) redaktor

NLA redaktori abil saab tegevusi muuta ja ümber seadistada ilma tüütu juhtkaadritega jäändamiseta. Seda kasutatakse tihti stseeni animeerimisel, et teha lihtsamini mastaapseid ja olulisi muutusi. Lisaks võimaldab see tegevusi ümber korraldada ja 'kihiti asetada', mis muudab lihtsamaks animatsiooni organiseerimise ja versioonikontrolli.

Trajektoorid


Trajektoorid on NLA kihtide süsteem. Kõige algelisemal tasemel aitab see ribasid organiseerida. Kuid tema abil saab ka liikumisi kihiti üksteisele liita, nii nagu pildiredaktoris piksleid üksteise peale: alumine esimesena, ülemine viimasena.



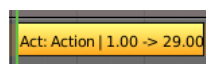
Ribad

On kolme sorti ribasid: tegevus, üleminek ja meta. Tegevused sisaldavad tegelikke juhtkaadreid, üleminekud arvutavad vahevorme tegevuste vahel ja meta grupeerib mitu riba üheks tervikuks.

Tegevusribade loomine

NLA redaktoris kasutamiseks tuleb tegevus kõigepealt muuta tegevusribaks. Seda tehakse, vajutades ikoonile  NLA nimekirjas oleva tegevuse kõrval. Võid ka valida

Menu: Add → Action



Tegevusriba.

Üleminekuribade loomine

Vali samal rajal kaks või rohkem riba

Menu: Add → Transition



Üleminekuriba.

Metaribade loomine

Kui on tarvis hulka ribasid koos liigutada, tuleb kasuks nende koondamine metaribaks, mis grupeerib nad kokku nii, et neid saab liigutada nagu ühtainust.

Menu: Add → Meta Strips



Vali Shift-klahvi all hoides kaks või enam riba. Ühenda nad metaribaks.

Ribade muutmine

Tegevusribade sisu saab muuta, kuid sa pead selleks olema 'näppimisrežiimis' (*Tweak mode*).

Menu: View → Enter Tweak Mode



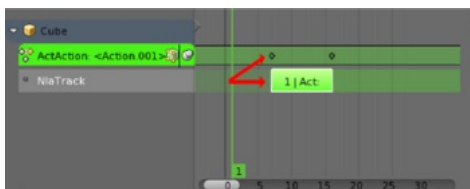
Riba NLA režiimis.

Riba
näppimisrežiimis.

Kui sa üritad muutmisrežiimis riba liigutada, siis sa märkad, et võtmed liiguvad ribaga kaasa. Teatud juhtudel on oluline, et võtmed püsiksid algsetes kaadrites, sõltumata sellest, kus riba parajasti asub. Selleks vajuta riba kõrval olevat 'lahtihaakimise' ikooni.



NLA riba kinnitatud võtmetega.



Liigutatud riba. NB! Võtmed liiguvad kaasa.



Lahti haagitud võtmed liiguvad tagasi algsetesse kaadritesse.

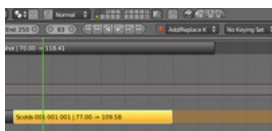
Kui oled riba muutmise lõpetanud, vali menüüst *View -> Exit Tweak Mode* (vaade -> lahku näppimisrežiimist). Vaikimisi on selleks tabulaator.

Ribade korduvkasutamine

Ühe tegevusriba sisu saab ajateljel mitmeid kordi toimuda. Et riba uuesti toimuma panna, vali riba ja siis menüüst

Menu: Edit → Duplicate Strips

Kui nüüd ükskõik millist rida näppida, siis muutuvad ka teised. Kui teisi riba koopiaid peale algse on näpitud, siis värvub algne rida punaseks.



Duplitseeritud rida.



Duplitseeritud rida muutmise ajal.

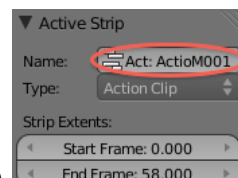
Algne rida.

Riba omadused

Riba omadusi saab muuta NLA päisest.

Menu: View → Properties

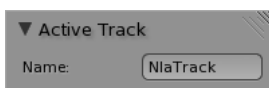
Ribade ümbernimetamine



Kõiki ribasid saab ümber nimetada riba omaduste jaotuses *Active Track* (aktiivne rada).

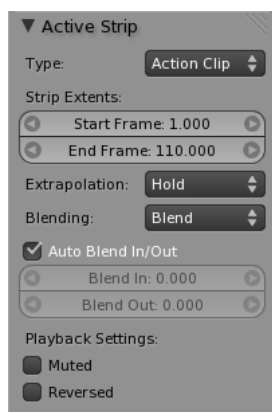
Aktiivne rada

See on rada, kuhu rida parajasti kuulub.



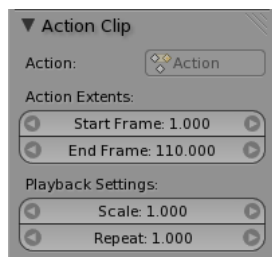
Aktiivne rida

Riba enda omadused. Tegevusriba võib olla kas liikumisklipp või üleminekuklipp. Pane tähele, et väljad *Strip Extents* (riba ulatus) piiritlevad ainult rida, mitte tegevust. Lisaks tähendab välja *Extrapolation* (ekstrapoleerimine) väärtus *Hold* (hoia) nii alguse kui ka järgneva hoidmist. Kui see on valitud, võib ta takistada eelmise klipi töötamist.

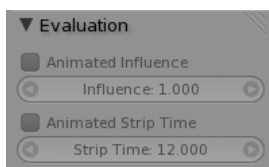


Aktiivne tegevus

Siin näidatakse riba 'objektiandmeid'. Nagu näiteks objekti muutmise väärtused.

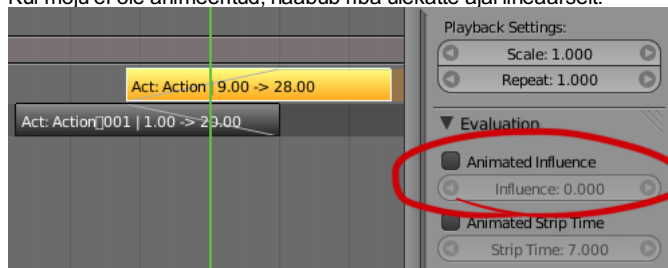


Väärtuse arvutamine



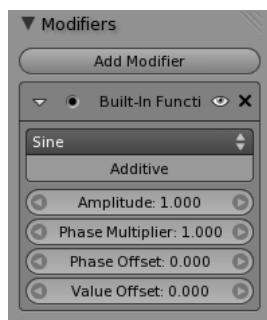
Määrab, kui palju ja mis aja jooksul riba mõjub.

Kui mõju ei ole animeeritud, hääbub riba ülekatte ajal lineaarselt.



Riba töötlemine

Sarnaselt oma sugulastele sõrestiku ja graafiku muutmiseks saab töötlemise abil lisada ribale kihiti erinevaid efekte. Ilmselgelt on see alles algus.



F-kõvera töötlevad

F-kõvera töötlevad sarnanevad objektitöötlevatele. Lähteandmeid rikkumata lisavad nad sellele mingi täiendava muutuse, mida saab keerulisemate efektide loomiseks hiljem igal ajal kohendada ja ka mitme töötleva mõju üksteise peale rakendada.

Töötleva lisamine

F-kõvera töötleva paneel asetseb omaduste (*Properties*) paneelil. Tähista kõver, valides ühe tema punkti või valides ta kanalite nimekirjast. Vajuta nupule **Add Modifier** (lisa töötleva) ja vali töötleva.

Töötlevate tüübid

Generaator

Algataja loob teguriteks lahutatud või laiendatud polünoomse funktsiooni. Need on matemaatilised põhivalemid, mis vastavalt kasutatavatele väärtustele kirjeldavad sirgeid, paraboole ja teisi keerulisemaid kõveraid.

Lisa (Additive)

Sellega mitte ei asendata kõvera olemasolevat töötlevat (vaikimisi), vaid lisatakse uus.

Polünoomi järk (Poly Order)

Määrab polünoomi järgu — st selle polünoomi 'x'-i suurima astme. (Kordajate arv = 1).

Kõvera kuju muutmiseks muuda kordajate (*Coefficient*) väärtusi. Lisainformatsiooni polünoomide kohta vaata [\[1\]](#) (inglise keeles).

Sisseehitatud funktsioonid

Need on täiendavad valemid, millel kõigil on samad valikud kõvera kuju juhtimiseks. Iga funktsiooni täpsemat kirjeldust vaata matemaatikakäsiraamatust.

- Siinus
- Koosinus
- Tangens
- Ruutjuur
- Naturaallogaritm
- Normaliseeritud siinus ($\sin(x)/x$)

Amplituud (Amplitude)

Muudab Y-skaalat

Faasikordaja (Phase multiplier)

Muudab X-skaalat

Faasi nihe (Phase Offset)

Muudab X-nihet

Väärtuse nihe (Value Offset)

Muudab Y-nihet

Ümbrik

Võimaldab sul muuta kõvera üldist kuju juhtpunktide abil.

Keskväärtus (Reference value)

Määrab ümbriku keskpunkti Y-väärtuse.

Miinimum (Min)

Alumine kaugus keskväärtusest, kus kehtib 1:1 vaikemõju.

Maksimum (Max)

Ülemine kaugus keskväärtusest, kus kehtib 1:1 vaikemõju.

Lisa punkt (Add Point)

Lisab juhtpunktide kogumi. Juhtpunktid tekivad parasjagu aktiivsesse kaadrisse.

Kaader Fra:

Määrab juhtpunkti kaadri numbri.

Miinimum (Min)

Määrab madalama juhtpunkti asendi.

Maksimum (Max)

Määrab kõrgema juhtpunkti asendi.

Tsüklid

Tsüklid võimaldavad sul lisada 2 või rohkema juhtpunktiga kõverale korduva liikumise. Seadeid saab määrata enne ja pärast kõverat.

Tsükli režiim (Cycle Mode)

Korda liikumist (Repeat Motion)

Kordab igas tsüklis kõverat samade väärtustega.

Korda nihkega (Repeat with Offset)

Kordab igas tsüklis kõverat, kuid nihutab igas tsüklis kõvera esimese punkti eelmise tsükli viimase punkti asukohta.

Korda peegelpildis (Repeat Mirrored)

Igas tsüklis peegeldatakse kõverat piki X-telge.

Enne/pärast tsüklit (Before/After Cycles)

Määra, mitu korda kõverat korratakse. Väärtus 0 kordab seda lõpmatult.

Müra

Lisab kõverale müra. See on kasulik, et tekitada animeeritud liikumises vaevumärgatavat või silmapaistvat juhuslikkust — näiteks kaamera värinat.

Segamisviis (Blend Type)

Asenda (Replace)

Lisab kõverale mürafunktsiooni vahemikus -0,5 kuni 0,5.

Liida (Add)

Lisab kõverale mürafunktsiooni vahemikus 0 kuni 1.

Lahuta (Subtract)

Lahutab kõverast mürafunktsiooni vahemikus 0 kuni 1.

Korruta (Multiply)

Korrutab kõvera väärtust mürafunktsiooniga vahemikus 0 kuni 1.

Skaala (Scale)

Määrab müra üldise skaala. Mida kaugemal on väärtus 0-st, seda väiksem on müra sagedus.

Tugevus (Strength)

Määrab mürafunktsiooni Y-skaala.

Faas (Phase)

Määrab mürafunktsiooni juhuslike arvude generaatori seemne.

Sügavus (Depth)

Määrab mürafunktsiooni detailsuse.

Python

Piirid

Piirab kõvera väärtused määratud X- ja Y-vahemikuga.

Minimaalne/maksimaalne X (Minimum/Maximum X)

Lõikab kõvera nende kaadrite pealt maha ja määrab minimaalsed väärtused nendes punktides.

Minimaalne/maksimaalne Y (Minimum/Maximum Y)

Kärbib kõvera väärtused vahemikku.

Astmeline

Annab kõverale astmelise väljanägemise, ümardades teatud kaadri vahemikus väärtusi allapoole.

Sammu suurus (Step Size)

Määrab, kui pikalt igat kaadrit hoitakse.

Nihe (Offset)

Kaadrite arv selle hetkeni, kui kaadreid hoidma hakatakse. Kasuta selleks, et tekitada '1-3' vs '5-7' ooteskeeme.

Kasuta alguskaadrit (Use Start Frame)

Piira töötajat nii, et ta mõjub ainult enne 'lõpp'kaadrit

Kasuta lõppkaadrit (Use End Frame)

Piira töötajat nii, et ta mõjub ainult pärast 'algus'kaadrit

Piirajate kasutamine animatsioonis

Piirajad võimaldavad omadusi (asukoha/pöoret/skaalat) juhtida kas lihtsate staatiliste väärtuste (näiteks "[piira asukoht](#)") või teise objekti (objektide) abil. Neid juhtobjekte kutsustakse "sihtmärkideks" (näiteks "[kopeeri asukoht](#)") abil.

Kuigi sääraseid piirajad võivad olla kasulikud ka staatilistes projektides, on nende peamine kasutus mõistagi animatsioon. Piirajad võivad animatsiooni mõjutada kahel viisil:

- sa saad juhtida objekti animatsiooni läbi piirajate poolt määratud sihtmärkide (see on üks kaudse animatsiooni vorm).
- Piirajate endi seadeid saab animeerida

Animatsiooni juhtimine piirajatega

See kehtib ainult sihtmärgiga (-märkidega) piirajate puhul. Sellisel juhul sihtmärgid juhivad piiraja valdajaobjekti omadusi ja seega sihtmärgi animeerimisel animeeritakse kaudselt ka valdajat.

Selline kaudselt "piiratud" animatsioon võib olla väga lihtne, nagu näiteks [asukoha kopeerija \(Copy Location\)](#), kus valdajaobjekt lihtsalt kopeerib oma sihtmärgi asukohta (vabalt valitud konstantse nihkega). Aga sa saad luua ka väga keerulist käitumist, nagu näiteks [tegevuspiiraja \(Action constraint\)](#) abil, mis töötab nagu [animatsiooni juhtija \(Animation Driver\)](#) tegevuste jaoks!

Me peame mainima ka klassikalist [alamobjektiga piirajat \(Child Of Constraint\)](#), mis loob objektide vahel ülem-alam suhte. Selline seos tekitab tõepoolest kaudse animatsiooni (ülemobjekti teisendamine mõjutab vaikselt kõiki tema alamobjekte). Ent alamobjektiga (Child Of) piiraja on väga oluline ka seetõttu, et ta võimaldab määrata objekti luu alamobjektiks ja seega kasutada selle animeerimiseks [skeletti](#)!

Tuleme tagasi lihtsa asukohta kopeerimise (Copy Location) näite juurde. See piiraja võib käituda kahte moodi:

- kui nupp Offset (nihe) on väljas (vaikimisi), siis on valdaja asukoht "täielikult" määratud piiraja sihtmärgi poolt — see tähendab, et mitte miski (peale pinus allpool olevate piirajate) ei saa muuta valdaja asukohta. Isegi mitte objekti animatsioonikõverad.
- Kui aga nupp Offset (nihe) on sisse lülitatud, siis on valdaja asukoht "suhteliselt" määratud piiraja sihtmärgi poolt. See tähendab, et valdaja asukohta väärtused on sihtmärgi asukohta suhtes nihkes. Ja neid valdaja asukohta suhtelisi väärtusi saab juhtida näiteks asukohta (Loc...) kõveratega (või tegevuste või NLA...) abil!

Näide

Kasutame piirajat Copy Location ja tema nuppu Offset (nihe). Näiteks saad sa panna valdaja (nimetagem teda *Kuuks*) tegema täiuslikke ringe ümber punkti (0.0, 0.0, 0.0) — kasutades näiteks Pythoni abil juhitud LocX/LocY animatsioonikõveraid (vaata [seda lehekülge](#)). Ja seejärel panna ta järgima sihtmärgi (mille nimi on... ma ei tea, näiteks *Maa*) asukohta, kui nupp Offset (nihe) on sisse lülitatud. Õnnitlused — sa modelleerisid just satelliidi (lihtsustatud) orbiidil ümber oma planeedi... Tee nüüd sedasama planeediga ümber tähe (millele sa võid, kui soovid anda nime *Päike*) — ja miks ka mitte tähega ümber galaktika...

Siin on väike animatsioon "päikesesüsteemist", mis on tehtud ülaltoodud tehnikat kasutades: [\[video link\]](#)

NB! See "päikesesüsteem" ei ole üldse realistlik (vale suurus, "maa" pöörleb ümber "Päikese" vales suunas jne).

Võid ([File:ManAnimationTechsUsingConstraintsExSolarSys.blend](#)) .blend-faali, millega see animatsioon on tehtud.

Piirajate mõju animeerimine

"Klassikalisem" võimalus on animeerida kõverate abil iga piiraja üksikuid väärtusi.

Sul on kasutada ainult kaks animatsioonikõverat (vaata ka [seda lehekülge](#)):

- sa võid animeerida piiraja mõju (Influence). Näiteks [ülaltoodud "päikesesüsteemis"](#) kasutasin ma seda selleks, et kinnitada kaamera alguses "Kuu" ja siis "Maa" külge ning lõpuks mitte millegi külge. Selleks oli vaja kahte Copy Location-piirajat sisselülitatud seadega Offset (nihe) ja mille Influence (mõju) muutus sujuvalt ühest väärtusest teiseks...
- Jaburana näiv võimalus on veel animeerida sellise piiraja, mille sihtmärgiks on skeleti luu, sihtmärgi tegelikku asukohta (juure ja tipu vahel) piki luud ("*0.0 tähendab "täiesti juurel" ja 1.0 "täiesti tipus"*).

Objektide liigutamine piki trajektoori

Animeerimisel on väga sageli vaja panna objektid liikuma mööda trajektoori. Mõtle keerulisele kaamera liikumisele või rongile rööbastel – ja enamus sõiduvahendeid võib samuti kasutada "nähtamatuid" radu! (Jalgratta keti lülid jne.) Kõiki neid liikumisi saaks mõistagi teha ka klassikaliste IPO-kõveratega, aga see oleks õudusunenägu! Palju lihtsam ja loomupärasem on defineerida vajaliku liikumise trajektor ja panna oma objekt(id) seda järgima.

Blenderil on objektide piki trajektoori liikuma panemiseks kaks erinevat piirajat. Need kasutavad erinevaid meetodeid, kuidas oma valdaja asukohta trajektoril määrata/animeerida.

Blenderis võib iga [kõver](#) olla trajektoriks. Kõverast saab trajektor siis, kui nupp Path Animation (trajektoranimatsioon) on paneelil Curve (Kõver) valitud. Sa ei pea aga ise sellega vaeva nägema: kui kõver on valitud "trajektoripiiraja" sihtmärgiks, tehakse seda automaatselt.

Sa võid ka lisada "trajektoori" ka otse (3D-vaates) menüüvalikuga Add » Curve » Path (lisa->kõver->trajektor). Sellega lisatakse sinu stseenile *kolmemõõtmeline* NURBS-kõver. See on oluline punkt: vaikimisi on Blenderi kõver 2-mõõtmeline, st paikneb täielikult ühel tasapinnal – mis ei ole tavaliselt soovitud trajektoori kuju. Et muuta tavaline kõver kolmemõõtmeliseks, lülita nupp 3D samal muutmispaneelil Curve and Surface (Kõver ja pind) välja.

Viimane omadus, mis on trajektoori jaoks oluline, on tema *suund*. Kolmemõõtmelistel trajektoridel märgitakse seda pisikeste noolekestega. Suunda saad sa muuta menüüst Curve » Segments » Switch Direction (kõver->lõigud->vaheta suunda) (või W2 NumPad).

Selle kohta, kuidas muuta trajektore/kõveraid, vaata täiendavalt [modelleerimise peatükist](#).

{{Note|Kujud kõverate järgij|Kui sa pigem soovid, et sinu objekti *kuju* järgiks trajektoori (nagu näiteks paberileht printeris), peaksid sa kasutama [kõveratöötajat](#)

Hierarhiline meetod

Vanemates Blenderi versioonides puudusid piirajad, millega objekti trajektoori järgima panna. Selle asemel kasutati teistsugust meetodit (mis on praeguseks iganenud, kuigi veel alles), mis põhineb ülem-alam hierarhial.

Selle meetodi kasutamiseks vali objekt, mis hakkab teekonda järgima. Seejärel vali klahviga ⇧ Shift kõver ja ava kiirklahviga CtrlP hierarhia menüü. Vali Follow Path (järgi trajektoori). Nüüd animeeritakse objekti mööda trajektoori.

Teekonna animeerimise seaded on kõvera omaduste paneeli alampaneelil Path Animation (trajektoranimatsioon).

Kaadrid (Frames)

Kaadrite arv, mille jooksul objekt teekonna läbib.

Arvutatud aeg (Evaluation Time)

Määrab, millises kaadris animatsioon parasjagu asub. Vaikimisi on see seotud globaalselt aktiivse kaadri numbriga, kuid sellele saab lisada võtmekaadreid, et objekti teekonda täpsemini juhtida.

Järgi (Follow)

Paneb piki trajektoori liikuvad objektid liikumise käigus kõvera suunas pöörduma.

Raadius (Radius)

Paneb piki trajektoori liikuvate objektide suuruse muutuma vastavalt määratud kõvera raadiusele. Vaata peatükki [kõvera eendamine](#)

Laste nihe (Offset Children)

Nihutab objektide asukohta animatsioonis vastavalt nende objektide ajanihke väärtusele. Viimase leiab objektipaneeli (Object Panel) alajaotusest Animation Hacks.

Trajektoripiiraja

Trajektoripiiraja Follow Path kasutab kõige "klassikalisemat" meetodit. Vaikimisi jalutab valdajaobjekt trajektoori läbi ainult ühe korra, alustades esimesest kaadrist ja tehes seda **100** kaadri jooksul. Sa võid määrata teistsuguse alguskaadri piiraja paneeli väljal Offset (nihe) ja muuta teekonna pikkust (kaadrites) paneeli Curve and Surface (kõver ja pind) välja (Frame (kaader) abil.

Sa võid oma objekti liikumist piki trajektoori palju täpsemalt juhtida võtmekaadrite abil või määrates trajektoori parameetritele Evaluation Time (arvutatud aeg) kiiruse (Speed) oma animatsioonikõvera. See kõver teisendab animatsiooni kaadri punktiks trajektoril **0.0** (alguspunkt) ja **1.0** (lõpp-punkt) vahel.

Põhjalikumat ülevaadet ja näiteid vaata [trajektoripiiraja leheküljelt](#).

Klammerduspiiraja

Teine täiuslikumat tehnikat kasutav meetod objektide trajektoril hoidmiseks on klammerduspiiraja (Clamp To). Et määrata, millises trajektoori punktis objekt hetkel asub, kasutab see *objekti asukohta* etteantud teljel. Seega selleks, et panna objekt liikuma piki trajektoori, tuleb sul animeerida (IPO-kõverate või muude kaudse animatsiooni vahendite abil) tema asukohta.

Sellest järeldub, et trajektoori pikkusel ei ole siin enam mingit tähtsust ja vaikimisi paikneb objekt staatiliselt mingis trajektoori punktis!

Põhjalikumat ülevaadet ja näiteid vaata [klammerduspiiraja peatükist](#).

Sissejuhatus füüsika simulatsiooni

See peatükk katab erinevaid keerulisemaid Blenderi võimalusi, mida kasutatakse sageli selleks, et simuleerida reaalseid füüsikalisi nähtusi, näiteks:

- Suits
- Vihm
- Tolm
- Riie
- Vesi
- Žele

[Osakeste süsteeme](#) (*Particle Systems*) kasutatakse paljude asjade, näiteks juuste, muru, suitsu ja parvede simuleerimiseks.

[Karvad](#) (*Hair*) on osakeste süsteemide alamhulk, mida kasutatakse kiuliste objektide, näiteks juuste, karvkatte, muru, sulgede jne puhul.

[Pehmed kehad](#) (*Soft Bodies*) on kasulik kõigi selliste asjade puhul, mis reageerides jõududele nagu gravitatsioon või tuul või põrkudes teiste objektidega, kipuvad painduma ja moonduma... Seda võib kasutada naha, kummi ja isegi riide jaoks, kuigi on olemas eraldi [riidesimulatsioon](#) (*Cloth*) spetsiaalselt riidesarnaste objektide jaoks.

[Jäigad kehad](#) (*Rigid Bodies*) simuleerivad objekte, mis on üsna jäigad.

On võimalik simuleerida [vedelikke](#) (*Fluids*) ja gaase, sealhulgas [suitsu](#) (*Smoke*).

[Jõuväljade](#) (*Force Fields*) abil saad muuta simulatsioonide käitumist.

Gravitatsioon

Gravitatsioon on üldine seade, mis rakendub ühtmoodi kõigile stseenis olevatele füüsikalistele süsteemidele ja asub stseeni kaardil. Selle väärtus jäetakse harilikult vaikimisi väärtuseks mis on -9.810 ühikut Z-telje suunas, mis sama nagu gravitatsioon reaalses maailmas. Selle väärtuse muutmine simuleerib madalamat või suuremat gravitatsioonijõudu.

Pane tähele, et sa saad välja kaalude (*Field Weights*) kaardil gravitatsiooni väärtust vähendada eraldi iga füüsikalise süsteemi kohta.

Force Fields - Jõuväljad

Jõuväljad pakuvad võimaluse lisada dünaamilistele süsteemidele liikumist. Jõuväljad võivad mõjutada nii [osakesi](#), [pehmeid kehasid](#) kui ka [riiet](#). Jõuväljad mõjutavad kõike automaatselt. Et eemaldada simulatsioon või osakeste süsteem nende mõju alt, vähenda lihtsalt sedasorti jõuvälja mõju selle välja kaalu paneelil.

- Välju võivad genereerida igat tüüpi objektid ja osakesed, aga ainult kõverobjektid tekitavad suunava (Curve Guides) tüüpi välju.
- Jõuvälju saab tekitada ka osakestest. Vaata [osakeste füüsikat](#)
- Objektid peavad jagama vähemalt ühte ühist kihti, et vastasmõju toimiks.

Mõju võib piirata osakestest kuni objektide gruppideni (vaata [osakeste füüsika](#) lehekülge).

Jõuvälja loomine

Mode: Objektirežiim

Panel: Objekti kontekst → füüsika alamkontekst → väljad

Hotkey: F7

Et luua üksikut jõuvälja, võid valida menüüst Add » Force Field (lisa->jõuväli) ja seejärel valida soovitud jõuvälja. See meetod loob tühiobjekti, millele on kinnitatud jõuväli.

Et luua välja olemasolevale objektile, vali objekt ja mine füüsika (Physics) alamkonteksti. Vali välja tüüp väljade (Fields) menüüst.

Väljadel on paljud valikud ühised, neid ühiseid valikuid selgitatakse keraja (Spherical) välja kohta.

Märkus

Pärast muudatusi (väljade (Fields) paneelil) või suunamise (põrke (Collision) paneelil) seadetes, pead arvutama uuesti osakeste, pehmete kehade või riide süsteemi (Free Cache - tühjenda vahemälu), see ei toimu automaatselt. Sa võid tühjendada vahemälu kõigil selekteeritud objektidel, valides CtrlB → Free cache selected (tühjenda valitud vahemälu).

Osakesed reageerivad igat tüüpi jõuväljadele (Force Fields), pehmed kehad aga ainult kerajatele (Spherical) väljadele, tuulele (Wind) ja keerisele (Vortex) (reageerivad ka harmoonilise liikumise (Harmonic) väljadele, aga mitte nii nagu vaja).

Ühised väljade seaded

Enamikul väljadel on samad seaded, kuigi nad käituvad väga erinevalt. Seadeid, mis on välja tüübile ainuomased, on kirjeldatud allpool. Suunaval kõveral ja tekstuuriväljadel on väga erinevad valikud.

Shape (kuju)

Väli on kas punktikujuline (Point) - suunast sõltumatu mõjuga või tasapinnaline (Plane) - muutumatu X-Y-suunal, muutuv ainult Z-suunas.

Strength (tugevus)

Välja mõju tugevus. See saab olla negatiivne või positiivne, muutes vastavalt jõu suunda. Jõuvälja tugevus muutub koos objekti skaleerimisega, mis võimaldab sul suurendada või vähendada kogu stseeni, säilitades samasuguse mõju objektide vahel.

Flow (vool)

Muudab mõjutava jõu õhuvoolu liikumiseks.

Noise (müra)

Lisab jõu tugevusele müra.

Seed (seeme)

Muudab juhusliku müra generaatori seemet.

Effect Point (mõjupunkt)

Saad lülitada sisse ja välja mõju osakese asukohale ja pöördele

Collision Absorption (neeldumine põrkel)

Jõud neeldub kokkupõrgetel objektidega.

Kahanemine (*Falloff*)

Siin saad määrata jõuvälja (kui kahanemise (Falloff) aste on suurem kui 0).

Sphere (kera)

Kahanemine on kõikides suundades ühesugune, nagu ka keral.

Tube (toru)

Kahanemise tulemuseks on torukujuline jõuväli.

Sellel tüübil saab muuta välja Radial falloff (külg-suunaline kahanemine) väärtust ja ka Minimum (minimaalset) ja Maximum (maksimaalset) vahemaad.

Cone (koonus)

Kahanemise tulemuseks on koonusekujuline jõuväli. Lisavalikud on samad, mis torukujulise (Tube) kahanemise puhul.

Z Direction (Z-suund)

Kahanemist saab määrata toimuma kas ainult positiivses Z-suunas, negatiivses Z-suunas või mõlemas.

Power (aste)

Kuidas jõuvälja tugevus muutub vastavalt kaugusele jõuväljast. Kui r on kaugus objekti keskpunktist, siis tugevus muutub vastavalt valemile $1/r^{\text{Power}}$. Kahanemine astmes 2 muudab jõuvälja valemiga $1/r^2$, mis on gravitatsioonilise tõmbe kahanemine.

Max Distance (maksimaalne kaugus)

Paneb jõuvälja mõjuma ainult ainult kindlast maksimumraadiusest seespool (näha lisarõngana ümber objekti).

Min Distance (minimaalne kaugus)

Vahemaa objekti keskpunktist kuni kauguseni, kus jõuväli mõjub täie tugevusega. Kui kahanemise väärtus on 0, siis ei tee see seade mitte midagi, sest väli mõjub täie tugevusega kuni maksimaalse kauguseni (Max Dist; või hoopis lõpmatuseni). Näha lisarõngana ümber objekti.

Väljade tüübid

Jõud (*Force*)

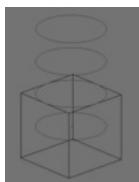
Jõud (Force) on väljadest kõige lihtsam. Tekitab konstantse jõu, mis on suunatud objekti keskpunkti poole (positiivne jõud) või sellest eemale (negatiivne jõud). Newtoni osakesed tõmbuvad negatiivse jõuga välja poole ja tõugatakse eemale positiivse jõuga välja poolt.

[Parvede \(Boids\)](#) jaoks saab positiivse jõuga välja kasutada *eesmärgina*, negatiivse jõuga välja aga *kiskjana*. See, kas parved otsivad või eemalduvad eesmärgist/kiskjatest, sõltub parvede füüsika seadetest.



Pilt 2b: Keraja
välja
indikaator

Tuul (*Wind*)



Pilt 3a: Tuule
(Wind)
indikaator

Tuul tekitab konstantse ühesuunalise jõu piki jõuobjekti kohalikku Z-telge. Jõu tugevus on nähtav ringide vahekauguse järgi.

Keerisväli (*Vortex Field*)



Pilt 3b:
Keerisvälja
(Vortex Field)
indikaator.

Keeris tekitab spiraalse jõu, mis väändub ümber objekti kohaliku Z-telje. Seda võib kasutada valamu äravoolu, tornaado või juuksekrusside tegemiseks.

Magnetväli (*Magnetic*)

See väli sõltub osakeste kiirusest. Simuleerib magnetvälja jõudu magnetiseeritud objektidele.

Harmoniline (*Harmonic*)

Jõuvälja allikaks on harmoonilise ostsillaatori (vedru, pendel) nullpunkt. Kui määrad seade Damping (summutamine) väärtuseks 1, peatub liikumine kohe, kui objektini on jõutud. See jõuväli on küllaltki eriline, kui rakendada teda osakestele.

Rest Length (pikkus puhkeolekus)

Määrab harmoonilise liikumise jõu pikkuse puhkeasendis.

Multiple Springs (mitu vedrut)

Põhjustab iga punkti mõjutamist mitme vedru poolt.

Tavaliselt mõjutab väljasüsteemi iga osake sihtmärksüsteemi iga osakest. Mitte aga harmoonilise välja puhul! Siin on iga sihtmärkosakese jaoks määratud kindel välja osake. Nii liiguvad osakesed teiste osakeste kohale, moodustades kujundeid.

[Õppetükk: Kujundeid moodustavad osakesed.](#)

Laeng (*Charge*)

See on samane kerajale väljale selle vahega, et ta muudab oma käitumist (külgetõmme/eemalletõukamine) olenevalt mõjutatavate osakeste laenguväljast (negatiivne/positiivne), nagu reaalsete laenguga osakeste puhul. Seega mõjutab see väli ainult osakesi, millel on samuti laenguväli (Charge) (vastasel juhul ei ole neil "laengut" ja seetõttu neid ei mõjutata)!

Lennard-Jones

See väli on väga lühikese ulatusega jõud, mille käitumise määrab mõjutavate osakeste ja mõjutatavate osakeste suurus. Osakeste kombineeritud suurusest väiksema vahemaa puhul on see väli tugeva tõukejõuga ja sellest vahemaast kaugemal tõmbejõuga. See üritab hoida osakesed üksteisest tasakaalustatud kaugusel. Osakesed peavad olema üksteisele küllalt lähedal, et see väli neid üldse mõjutaks.

Osakestel võivad olla näiteks nii laeng kui ka Lennard-Jonesi väli - mis on arvatavasti midagi tuumafüüsikutele meie hulgas.

Tekstuur väli (*Texture Field*)

Sa saad tekstuuritud jõuväljade abil luua ise suvalise keerukusega jõuvälju, mille jõud on kolmes suunas kodeeritud värvidega. Punane märgib x-telge, roheline y-telge ja sinine on z-telje jaoks (nagu värvid kordinaattelgedel 3D-aknas). Väärtus 0.5 tähendab, et jõudu ei ole, suurem väärtus kui 0.5 kiirendust telje negatiivsel suunal (nagu -Z), väiksem väärtus kui 0.5 kiirendust telje positiivsel suunal (nagu +Z).

Texture mode (tekstuuri režiim)

See määrab, mismoodi jõuvektor tekstuurist tuletatakse.

RGB

Kasutab värvi komponente otse jõuvektori komponentidena värvi poolt näidatud suunas. Sul on selle jaoks vaja RGB tekstuur, näiteks pilti või värvikaarti. Seega ei ole sulandatud (Blend) tekstuur ilma värviüleminekuta piisav.

Gradient

Arvutab jõuvektori vastavalt teksturi (halltoonide) intensiivsuse 3D-gradiendile. Gradiendivektor on alati suunatud suureneva heleduse suunas.

Curl (keerd)

Arvutab jõuvektori kolmemõõtmelise RGB teksturi keerdumise järgi (RGB-vektorite pöördumine). See töötab samuti ainult värvi tekstuuridega. Seda võib näiteks kasutada selleks, et luua kena välimusega turbulentne jõud värvilisest pilvetekstuurist koos Perlini müraga.

Nabla

See on teksturi gradient- ja keerurežiimide puhul osatuletiste arvutamiseks kasutatav nihe.

Use Object Coordinates (kasuta objektkoordinaate)

Kasutab emiteeriva objekti koordinaate (ja pööret ja suurust) tekstuurikoordinaadistikuna, mida osakesed kasutavad. Võimaldab luua liikuvaid jõuvälju, mille koordinaadid on seotud objekti asukoha koordinaatidega.

Root Texture Coordinates (juure tekstuurikoordinaadid)

See on kasulik karvade puhul, kuna kasutab osakese juure asukoha jaoks arvutatud tekstuurjõudu kõigi karva osade jaoks.

2D

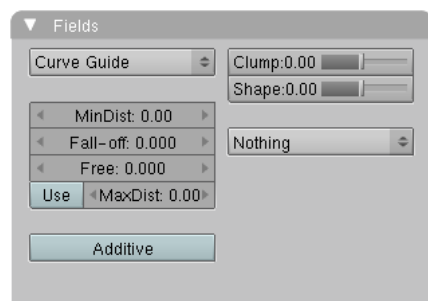
Nupp 2D eirab osakeste Z-koordinaati ja kasutab ainult osakeste x ja y koordinaate nende tekstuurikoordinaatidena.

Pea meeles, et ainult protseduurilised tekstuurid on päriselt 3D.

Näited

- Üksik värvitekstuur 0.5/0.0/0.5 loob jõu positiivse y-telje suunas, näiteks karvad on suunatud y-telje suunas.
- Värviüleminekuga teksturi saab kasutada, et luua jõu "tasapinda". Näiteks kui teksturi väärtus vasakul pool on 0.5/0.5/0.5 ja paremal pool 1.0/0.5/0.5, saad jõutasapinna, mis on risti XY-tasandiga (ehk paralleelne Z teljega). Kui kasutad koordinaatide allikana objekti, siis saad kasutada objekti osakeste lükkamiseks.
- Animeeritud puidutekstuuri saab kasutada lainetava liikumise loomiseks.

Suunav kõver (*Curve Guide*)



Pilt 4a: Suunava kõvera (Curve Guide) väli.

Kõverad (Curves) saavad olla suunava kõvera (Curve Guide) välja allikaks. Selle abil saab juhtida osakesi mööda kindlat trajektoori, pehmeid kehasid nad ei mõjuta. Tüüpiline kasutusstsenaarium oleks liigutada punast vereliblet veresoones või animeerida osakeste voogu mootoris. Sa võid kasutada suunavat kõverat, andmaks kuju teatud karvadele - kuigi seda ei kasutata enam nii tihti, kuna nüüd on meil olemas [osakeste režiim](#). Kuna sa saad kõveraid animeerida pehmete kehadena või mõnel teisel tavalisel viisil, saad sa ehitada väga keerukaid animatsioone, säilitades samal ajal täpse kontrolli tulemuse üle ja hoides simulatsiooni aja minimaalse.

Valik Curve Follow (järgi kõverat) ei tööta osakestel. Selle asemel pead sa seadma välja Angular Velocity (nurkkiirus) füüsika paneelil osakese (Particle) alamkontekstis väärtuseks Spin (pöörlemine) ja jätmata pöörlemise konstantseks (ära lülita sisse valikut Dynamic (dünaamiline)).

Suunavad kõverad (Curve Guide) mõjutavad kõiki osakesi samal kihil, olenemata nende kaugusest kõverast. Kui sul on kihil mitmed suunajad, siis nende väljad liituvad üksteisega (nagu sa võib-olla õppisid füüsika tunnis). Aga sa saad limiteerida nende mõju raadiust:

Minimum Distance (vähim kaugus)

Vahemaa kõverast kuni kauguseni, kus jõuvali mõjub täie tugevusega. Kui sul on välja Fall-off (kahanemine) väärtuseks 0, siis ei tee see näitaja mitte midagi, sest väli mõjub täie tugevusega kuni maksimaalse kauguseni (Max Dist) (või lõpmatult kaugele). Minimaalne kaugus (MinDist) on 3D-aknas näha ringidena kõvera otstes.

Free (vaba)

See osa osakese elueast, mida kõvera jaoks ei kasutata.

Fall-off (kahanemine)

See seade määrab suunaja tugevuse minimaalse (MinDist) ja maksimaalse (MaxDist) kauguse vahel. Kahanemise (Fall-off) väärtus 1 tähendab lineaarset seost.

Osake järgib suunavat kõverat (Curve Guide) selle eluea jooksul, tema kiirus sõltub eluea ja tee pikkusest.

Additive (liituv)

Kui sa kasutad seadet Additive (liituv), siis sõltub ka osakeste kiirus kahanemise (Fall-off) väärtusest.

Weights (kaalud)

Kasuta kõvera kaalu, et määrata osakese mõju mõõda kõverat.

Maximum Distance/Use Max (maksimaalne kaugus/kasuta maksimaalset)

Maksimaalne mõjuraadius. See on näha lisarõngana ümber kõverobjekti.

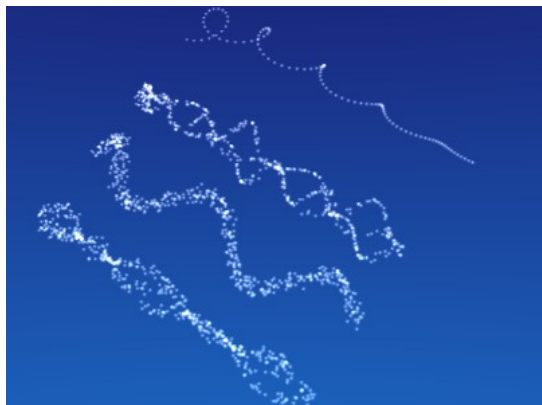
Teised seaded määravad jõuvalja kuju piki kõverat.

Clumping Amount (kogunemise määr)

Osakesed kogunevad kõvera lõpus (1) või eralduvad üksteisest (-1).

Shape (kuju)

Määrab, mis kujuliselt osakesed kogunevad. +0.99: Osakesed kogunevad kõvera lõpus. 0: lineaarne liikumine piki kõverat. -0.99: osakesed kogunevad kõvera alguses.



Pilt 4b: Väände (Kink) valikud suunaval kõveral. Vasakult paremale: Radial (radiaalne), Wave (laine), Braid (palmik), Roll (pöörlemine).

[Animatsioon](#)

Rippmenüüga Kink (vääne) saad sa varieerida jõuvalja kuju:

Curl (lokk)

Mõju raadius sõltub emiteerija ja kõvera vahelisest vahemaast.

Radial (radiaalne)

Kolmemõõtmeline seisev laine.

Wave (laine)

Kahemõõtmeline seisev laine.

Braid (palmik)

Palmik.

Roll (pöörlemine)

Ühemõõtmeline seisev laine.

Ei ole lihtne saada vaid vorme kirjeldada, loodan, et need on küllalt arusaadavalt näha illustratsioonil (*Pilt 4b*).

Frequency (sagedus)

Nihke sagedus.

Shape (kuju)

Määrab nihke kas algusesse või lõppu.

Amplitude (amplituud)

Nihke ulatus.

Parv (*Boid*)

...

Turbulents (*Turbulence*)

Loob juhusliku turbulentsi 3D-müra abil.

Size (suurus)

Määrab müra skaala.

Global (globaalne)

Muudab müra suuruse ja tugevuse sõltuvaks maailma koordinaatidest (mitte selle objekti koordinaatidest, millele see lisatud on).

Takistus (*Drag*)

Takistus (*Drag*) on jõud, mis töötab osakeste liikumisele vastu, aeglustades neid.

Linear (lineaarne)

Takistus on proportsionaalne kiirusega.

Quadratic (ruutsõltuvus)

Takistus on proportsionaalne kiiruse ruuduga.

Lingid

- [Tuule ja suunava jõu täiendused 2.48](#)
- [Osakeste seaded ja juhid \(v2.40\)](#)

Kokkupõrked

[Osakesed](#), [pehmed kehad](#) ja [riideobjektid](#) võivad põrkuda võreobjektidega. [Parved](#) püüavad vältida kokkupõrke (Collision) objekte.

- Kokkupõrke toimimiseks peavad objektid jagama vähemalt ühte ühist kihti.
- Sa võid toimet osakestele piirata osakestest grupi objektidega ([väljamõju](#)) paneelil *Field Weights*).
- Pehme kehade *suunamine* on väga keeruline ja nad kipuvad sageli põrkuvatest objektidest läbi tungima.
- Karvad (Hair) ignoreerivad suunavaid objekte (kuid sa võid need animeerida pehmete kehadena, mis võtavad põrget arvesse).

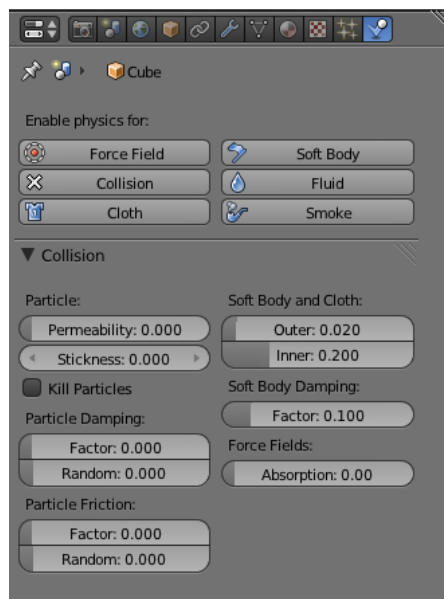
Kui muudad objektil põrke seadeid, pead sa uuesti arvutama osakeste, pehmete kehade või riide süsteemi (tühjenda vahemälu, Free Cache), see ei toimu automaatselt. Sa võid tühjendada vahemälu kõigil selekteeritud objektidel, valides CtrlB → Free cache selected (tühjenda valitud vahemälu).

Mode: Objektirežiim

Panel: Objekti kontekst Object → füüsika alamkontekst Physics → kokkupõrge (Collision)

Hotkey: F7

Valikud



Pilt 1: Füüsika alamkonteksti Physics kokkupõrke paneel Collision.

Permeability (läbistatavus)

Osa osakestest läbib võret. Seda saab animeerida objekti IPOdega kanalis Perm.

Stickiness (kleepuvus)

Kui palju osakesi jääb objekti külge kinni.

Kill Particles (tapa osakesed)

Kustutab osakesed kokkupõrkel.

Damping Factor (summutusfaktor)

Summutus põrke jooksul (ei sõltu osakeste kiirusest).

Random (juhuslik)

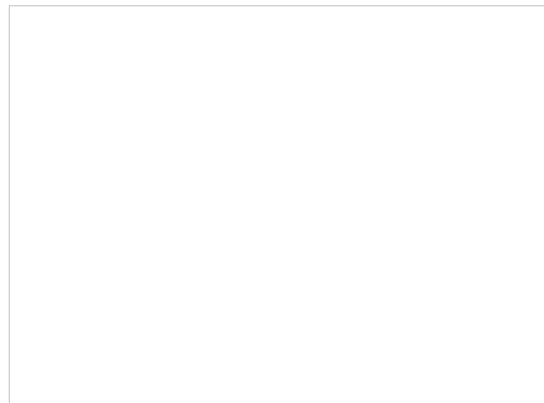
Summutuse juhuslik varieeruvus.

Friction Factor (hõõrdetegur)

Hõõrdumine liikumisel mööda pinda.

Random (juhuslik)

Hõõrdumise juhuslik varieeruvus.



Pilt 1b: Pehme keha tüüpi kokkupõrge tasandiga

Pehme keha ja riide vastastikune mõju

Outer (väline)

Välise pörketsooni suurus.

Inner (sisemine)

Sisemise pörketsooni suurus (vetrumise ulatus).

Välimine ja sisemine külg on määratud küljenormaaliga (*Pilt 1b*).

Damping Factor (summutustegur)

Summutus pörke jooksul.

Pehme keha pörkeid on raske täiuslikuks saada. Kui üks objektidest liigub liiga kiiresti, tungib pehme keha võrest läbi. Vaata ka [pehmete kehade](#) peatükki.

Vastastikmõju jõuväljadega

Absorption (neeldumine)

Tõukaja mõjub ka mõjutajatele. Sa saad määrata mõned pörkuvaid/suunavad objektid, mis tõukavad kõrvale kindla osa mõjutaja jõust, neeldumise (Absorption) väärtuse abil. 100% neeldumise tulemuseks on, et jõud ei pääse läbi pörkuva/tõukava objekti. Kui sul on 3 pörkeobjekti, millel neeldumised on 10%, 43% ja 3%, annab see tulemuseks 50% neeldumist ($100 \times (1 - 0.1) \times (1 - 0.43) \times (1 - 0.03)$).

Näited



Pilt 2: Eemaletõugatud objektid

Siin on metaobjekt (Meta), mille tipud on duplitseeritud osakeste süsteemiks. Osakesi emiteeritakse allapoole ja kuupvõre tõukab neid eemale.

Vihjed

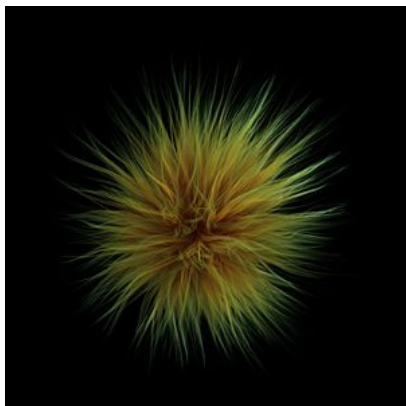
- Veendu, et võre pinna normaalid oleksid suunatud osakeste/punktide poole, et eemaletõukamine oleks õige.
- Karva (Hair) tüüpi osakesed reageerivad jõuväljadele otse, seega kui kasutad jõuvälja lühikeses ulatuses, ei ole pörget tingimata tarvis.
- Karva (Hair) tüüpi osakesed väldivad neid emiteerivat võret, kui muudad neid osakeste (Particle) režiimis. Seega saad vähemalt karvu pörkega modelleerida.

Osakesed

Osakesed on suur hulk (harilikult tuhandeid) elemente, mida emiteeritakse võreobjektist. Iga osake võib olla valgustäpp või võre ja osakesed võivad olla omavahel ühendatud või dünaamilised. Nad võivad reageerida paljudele erinevatele mõjudele ja jõududele ja neil on oma eluiga. Dünaamilised osakesed võivad kujutada tuld, suitsu, udu ja muid selliseid asju, nagu näiteks tolm või maagilised loitsud.

[Karva](#) (Hair) tüüpi osakesed on tavaliste osakeste alamklass. Karvasüsteemid moodustavad kiud, mis kujutavad juukseid, karusnahka, muru ja harjaseid.

Sa näed osakesi osakestetõttlejana (Particle Modifier), kuid neid seadistatakse objektikonteksti osakeste alamkontekstis.



Pilt 1: Osakestest tehtud karusnahk ([blend-fail](#)).

Harilikult voolavad osakesed oma võrest ruumi. Nende liikumist võivad mõjutada mitmed asjad, nende hulgas:

- Algkiirus võrest väljudes.
- Emiteerija (tipp, külg või objekt ise) liikumine.
- Liikumine vastavalt "gravitatsioonile" või "õhutakistusele".
- Teiste jõuväljade, näiteks tuule või keeriste mõju, või juhtimine mööda kõverat.
- Vastastikmõju teiste objektidega, näiteks kokkupõrked.
- Osaliselt intelligentsed parve liikmed (karjad, parved jms), mis suhtlevad teiste oma parve liikmetega, kui üritavad jõuda sihtmärgini või vältida kiskjaid.
- Sujuv liikumine pehme keha füüsikaga (ainult karvaosakeste süsteemides).
- Või isegi käsitsi muutmine [sõrestike](#) abil.

Osakesi võib renderdada:

- [Halodena](#) (leegid, suits, pilved).
- Võredena, mida saab omakorda animeerida (näiteks kalad, mesilased jms). Nendel juhtudel iga osake "kannab" teist objekti.
- [Kiududena](#) (*Strands*) ([karvade](#), [karusnaha ja rohu](#)) jaoks; osakese kogu trajektoor on näha kiuna. Neid kiudusid saab 3D-aknas manipuleerida (kombineerida, lisada, lõigata, liigutada jne).

Igal objektil võib olla mitu osakeste süsteemi. Iga osakeste süsteem võib sisaldada kuni 100 000 osakest. Kindlatel osakeste tüüpidel (karvad ja võtmeosakesed) võib olla kuni 10 000 alamosakest iga osakese kohta (alamosakesed liiguvad ja käituvad enam-vähem nagu nende vastavad ülemosakesed). Praktilisteks piirideks on su mälu maht ja kannatus.

Mitteühilduvus varasemate versioonidega

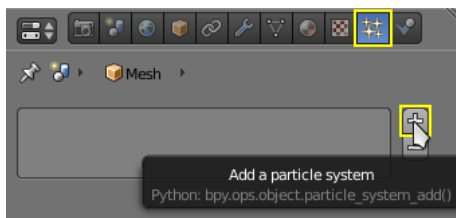
"Vana" osakeste süsteemi, mida kasutati kuni versioonini 2.45 (kaasa arvatud), ja "uue" osakeste süsteemi vahel on mitmeid erinevusi. Nüüd on võimalikud paljud asjad, mida ei saanud teha vana süsteemiga. Uus süsteem on vana süsteemiga kokkusobimatu. Kuigi Blender üritab konverteerida vanu osakeste süsteeme, töötab see ainult osaliselt. Vana süsteem meenutab kõige rohkem uut emiteerija (Emitter) süsteemi (loe edasi, et teada saada, mis see on). Kui kasutad Blender 2.45 või vanemat, [loe vanemat dokumentatsiooni siit](#).

Tööjärjekord

Standardne osakestega töötamise protsess on järgmine:

1. Loo võre, mis hakkab osakesi emiteerima.
2. Loo üks või mitu osakeste süsteemi, mida võre hakkab emiteerima. Sageli mitu osakeste süsteemi mõjuvad vastastikku ja liituvad, et saavutada summaarne soovitud efekt.
3. Sobita iga osakeste süsteemi seaded, et saada soovitud tulemus.
4. Animeeri baasvõre ja teised osakeste võred, mis stseenis osalevad.
5. Määra ja kujunda osakeste trajektoor ja voog.
6. [Karvaosakeste](#) süsteemide puhul: Vooli emiteerija voog (näiteks löika juuksed paraja pikkuseni ja kammi).
7. Tee lõplik renderdus ja füüsika simulatsioonid ning kohenda, kui vaja.

Osakeste süsteemi loomine

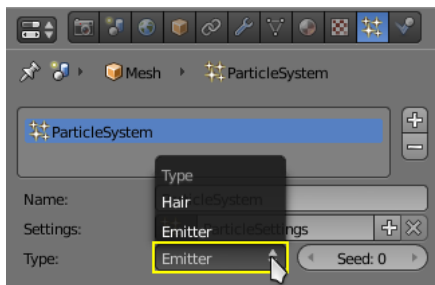


Pilt 2: Osakeste süsteemi lisamine.

Et lisada objektile uut osakeste süsteemi, mine objekti seadete redaktori osakeste (*Particles*) kaardile ja klõpsa väikest + nuppu. Objektile võib olla mitu osakeste süsteemi.

Igal osakeste süsteemil on eraldi seaded. Need seaded võivad olla jagatud erinevate osakeste süsteemide vahel, nii ei pea käsitsi kopeerima iga seadet, vaid sama efekti saab kasutada mitmel objektil. Kui kasutada juhuslikkust (Random), muudetakse omadusi juhuslikult natukene nii, et kui kasutad samu seadeid, paistavad need natuke erinevad.

Osakeste süsteemide tüübid



Pilt 3: Osakeste süsteemide tüübid

Kui oled loonud osakeste süsteemi, täitub omaduste (Property) aken paljude paneelide ja nuppudega. Aga ära satu paanikasse! On kaks erinevat tüüpi osakeste süsteemi ja sa saad nende vahel liikuda rippmenüüga Type (tüüp):

Emitter (emiteerija)

See on vana süsteemiga kõige sarnasem. Sellistes süsteemides emiteeritakse valitud objektist alates alguskaadrist (Start) kuni lõppkaadri (End) osakesi, millel on kindel eluiga.

Karvad

See süsteemi tüüp renderdatakse kiududena ja tal on mõned väga erilised omadused: teda saab 3D-aknas reaalaajas muuta ja lisaks saad kiude animeerida [riidesimulatsiooni](#) (Cloth Simulation) abil.

Seaded osakeste süsteemi paneelil on iga süsteemi tüübi puhul osaliselt erinevad. Näiteks *Pildil 3* on need näidatud ainult emiteerija (Emitter) süsteemi tüübi jaoks.

Ühised valikud

Igal süsteemil on samad põhiseaded, kuid valikud nende sees erinevad sõltuvalt kasutatavast süsteemist. Need seadete kogumid on:

Emission (emissioon)	Seaded osakeste esialgselt jaotumiseks emiteerijal ja viis, kuidas nad stseeni sünnivad.
Cache (puhver)	Et kiirendada reaktsiooniga reaalaajas ja vältida mittevajalikke taasarvutamisi, saab osakese andmeid säilitada mälus või salvestada kettale.
Velocity (kiirus)	Osakeste algkiirus.
Rotation (pöörlemine)	Osakeste pöörlemine.
Physics (füüsika)	Kuidas osakeste liikumine käitub.
Render (renderdamine)	Renderdamise seaded.
Display (esitus)	Reaalaajas kuvamine 3D-vaates.
Children (alamosakesed)	Täiendavate alamosakeste juhtimine.
Field Weights (välja kaalud)	Välise jõudude mõjutugevused.
Force Field Settings (jõuväljade seaded)	Tekitab osakesele jõu välja.
Vertex Groups (tipugrupid)	Erinevate seadete mõjutamine tipugruppide abil.

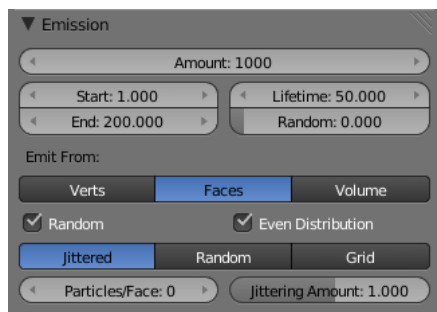
Lingid

- [Õppetükid](#)
- [Füüsika puhverdamine ja eeltöötlus](#)
- [Osakeste süsteemi ümberehituse dokumentatsioon](#)
- [Mõtted osakeste süsteemi ümberehituse koodist](#)
- [Staatiliste osakestega karusnahkade kollektsioon](#)

Osakeste emissioon

Emiteeriv süsteem (Emitter) töötab just nii, nagu nimigi ütleb: ta emiteerib/toodab kindla aja jooksul osakesi. Sellistes süsteemides emiteeritakse valitud objektist alates alguskaadrist (Start) kuni lõppkaadriini (End) osakesi, millel on kindel eluiga. Need osakesed renderdatakse vaikeväärtusena nagu [halod](#), aga sa võid selliseid osakesi renderdada ka objektidena (olenevalt osakeste süsteemi renderdusseadetest, vaata peatükki [visualiseerimine](#)).

Valikud



Pilt 2a: Osakeste emiteerimise seaded.

Nupud emissiooni (Emission) paneelil määravad, mismoodi osakesi ajas genereeritakse:

- Amount (kogus)**
maksimaalne ülemosakeste hulk, mida simulatsioonis kasutatakse.
- Start (algus)**
osakeste emissiooni alguskaader. Võid seada negatiivse väärtuse, mis võimaldab alustada simulatsiooni enne tegelikku renderdamist.
- End (lõpp)**
osakeste emissiooni lõppkaader.
- Lifetime (eluiga)**
osakeste eluiga (kaadrites).
- Random (juhuslik)**
osakeste eluea juhuslik varieeruvus. Lühim võimalik eluiga on $eluaeg \times (1 - juhuslik)$. Väärtused üle 1 ei ole lubatud.
Näiteks eluea (Lifetime) vaikeväärtusega 50 ja juhusliku varieeruvusega (Random) 0.5 luuakse osakesed, mille eluiga on alates 50st kaadrist kuni $50 \times (1.0 - 0.5) = 25$ kaadriini ja juhusliku varieeruvusega (Random) 0.75 luuakse osakesed, mille eluiga on 50 kaadrit kuni $50 \times (1.0 - 0.75) = 12.5$ kaadrit.

Emissiooni asukoht

Emit From (emissiooni lähtekoht) parameetrid määravad, kuidas ja kust osakesed emiteeritakse, andes täpse kontrolli nende jaotuse üle. Sa võid kasutada tippude gruppe, et hoida emissiooni kindlates piirides - seda tehakse tipugruppide (Vertexgroups) paneelil.

- Verts (tipud)**
Emiteerib osakesi võre tippudest.
- Faces (küljed)**
Emiteerib osakesi võre külgede pinnalt.
- Volume (maht)**
Emiteerib osakesi suletud võre sisemusest.

Jaotuse seaded

Need seaded kontrollivad, kuidas osakeste emiteerimine on läbi emissioonikohtade jaotunud.

- Random (juhuslik)**
emiteeriva elemendi indeksid käiakse läbi juhuslikus järjekorras, mitte lineaarselt (üksteise järel).

Külgede ja mahu puhul ilmuvad lisavalikud:

- Even Distribution (ühtlane jaotus)**
Osakesed jaotatakse võrdeliselt elementide pindaladele: näiteks väiksed elemendid emiteerivad vähem osakesi kui suured ja nii on osakeste tihedus võrdne.

- Jittered (värinaga)**
Osakesed paigutatakse emiteerivatele elementidele ebaühtlase intervalliga.

Particles/Face (osakesi külje kohta)
Emissioonide arv külje kohta (0 = automaatne).

JitteringAmount (värina määr)

Kui palju ebaühtlust asukohtade leidmisel rakendatakse.

- Random (juhuslik)**
Osakesed emiteeritakse elementide juhuslikest kohtadest.

- Grid (ruudustik)**
Osakesed seatakse 3D-ruudustikku ja osakesed elementide lähedal/sees säilitatakse.

Invert Grid (pööra ruudustik ümber)

Vahetab ära selle, mida loetakse objektiks ja mida mitte.

Hexagonal (kuusnurkne)

Kasutab kuusnurkset ruudustikku nelinurkse asemel.

Resolution (lahutus)

Ruudustiku tihedus.

Random (juhuslik)

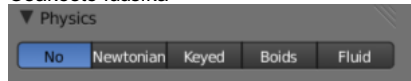
Lisab ruudustiku asukohtadele juhusliku nihke.



Et võre sisemusest osakesi emiteerida, peab su võre olema vettpidav.

Mõned töötlejad (näiteks servade lõhestaja (Edge Split) lõhuvad pinda, misjuhul emissioon võre sisemusest ei tööta korralikult!

Osakeste füüsika



Pilt 1: füüsika tüübid

Osakeste liikumist saab juhtida erinevate meetoditega. Osakeste füüsikas on viis erinevat süsteemi:

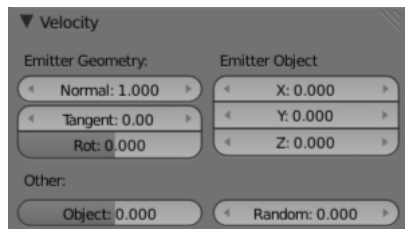
- None* (puudub) See ei anna osakestele mingit liikumist, mistõttu ei kuulu nad ühtegi füüsikalisse süsteemi.
- [Newtonian \(Newtoni\)](#) Liikumine vastavalt füüsikaseadustele.
- [Keyed \(võtmeosakesed\)](#) Dünaamilised või staatilised osakesed, kus (animeeritud) sihtmärkideks on teised osakeste süsteemid.
- [Boids \(parved\)](#) Osakesed piiratud tehisintellektiga, mis muuhulgas võimaldab käitumise ja reeglite programmeerimist - ideaalne linnu- või kalaparvede või kiskjate-saakloomade simuleerimiseks.
- [Fluid \(vedelik\)](#) Liikumine vastavalt vedeliku seadustele (põhineb silutud osakeste hüdrodünaamikal).

Lisameetodid osakeste liigutamiseks:

- Pehme keha animatsioon (ainult karvaosakeste süsteemide jaoks).
- Jõuväljade abil ja mööda kõveraid.
- Sõrestike abil.

Siin käsitleme ainult osakeste füüsikat kitsamas tähenduses, st füüsika paneeli seadeid.

Kiirus (*Velocity*)



Pilt 3: Algiirus

Osakeste algiirust saab seadistada erinevate parameetrite kaudu, olenevalt osakeste süsteemi tüübist (vaata osakeste süsteemi kaarti). Kui osakeste süsteemi tüübiks on emiteerija või karvad, siis järgnevad parameetrid annavad osakestele algiiruse vastavates suundades...

Emiteeriv geomeetria (*Emitter Geometry*)

Normal (pinnanormaal)

Emiteerija pinnanormaalid (st laseb pinnanormaalidel anda osakestele algse kiiruse).

Tangent (puutuja)

Puutuja suund annab osakestele algse kiiruse.

Rot (pööra)

Pöörab pinna puutujat.

Emiteeriv objekt (*Emitter Object*)

Align X,Y,Z (joonda X, Y, Z)

Annab algiiruse X-, Y- ja Z-teljel.

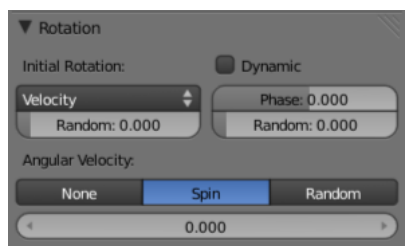
Object (objekt)

Emiteeriva objekti liikumine (st objekt annab osakestele algse kiiruse).

Random (juhuslik)

Lisab algiirusele juhuslikke variatsioone. Sa saad kasutada tekstuuri ainult väärtuse muutmiseks - vaata "emissiooni juhtimine, vastastikmõju ja aeg".

Pöörlemine



Pilt 4: Osakeste pöörlemine

Need näitajad määravad, kuidas üksikud osakesed oma teekonna jooksul pöörlevad. Et osakese pöörlemist visualiseerida, peaksid sa valida visualisatsiooni paneelil (Visualization) visualisatsiooni tüübiks telje ja suurendama parameetrit Draw Size (joonistamise suurus).

Initial Rotation Mode (algse pöörlemise režiim)

Seab osakese algse pöörlemise, joondades tema x-telje:

None (puudub)

Globaalse x-telje suunas.

Normal (pinnanormaal)

Emiteerija pinnanormaali suunas.

Velocity (kiirus)

Osakese algse kiiruse suunas.

Global X/Global Y/Global Z (globaalne X/Y/Z)

Ühe globaalse telje suunas.

Object X/Object Y/Object Z (objekt X/Y/Z)

Emiteeriva objekti ühe telje suunas.

Random (juhuslik)

Muudab pöörlemise juhuslikuks.

Dynamic (dünaamiline)

Kui see on sisse lülitatud, siis antakse osakestele ainult soovitud algne pöörlemine ja nurkkiirus ja jäetakse ülejäänu füüsika hooleks. Osakesed muudavad seejärel oma nurkkiirust, kui põrkuvad teiste objektidega (nagu reaalses maailmas põrkuvate pindade vahelise hõõrdumise tõttu). Muudel juhtudel on nurkkiirus alati ette antud (st pöörlemine on määratud dünaamiliseks/konstantseks).

Phase (faas)

Algne keerlemisfaas

Random (juhuslik)

Lisab faasile juhusliku varieeruvuse.

Angular Velocity (nurkkiirus)

Nurkkiiruse suurus, rippmenüü määrab nurkkiiruse telje.

None (puudub)

Nullvektor (pöörlemine puudub).

Spin (spinn)

Osakese kiirusvektor.

Random (juhuslik)

Juhuslik vektor.

Kui kasutad suunavat kõverat ja soovid, et osakesed kõverat järgiksid, pead seadma nurkkiiruse spinniks ja jätma pöörlemise konstantseks (st ära lülita sisse dünaamikat). Kõvera järgimine ei toimi osakeste puhul.

Ühised füüsika seaded

Size (suurus)

Määrab osakeste suuruse.

Random Size (juhuslik suurus)

Varieerib osakeste suurust juhuslikult.

Mass (mass)

Määrab osakeste massi.

Multiply mass with particle size (korruta mass osakese suurusega)

Teeb suurematele osakestele suurema massi.

Füüsika puudub

Algul võib füüsika tüüp, mis paneb osakese tegema mitte midagi, tunduda natuke imelik, kuid see võib mõnikord väga kasulikuks osutuda. Füüsika puudumine paneb osakesed kleepuma oma emiteerija külge kogu nende eluajaks. Näiteks on siin kasutatud algkiirust selleks, et anda kiirus osakestele, mida mõjutab selle füüsika tüübiga harmooniline mõjutaja, kui tema mõju osakestele lõpeb.

Veelgi enam, on väga mugav, kui su käsutuses on osakesed (millest nii sündimata kui ka surnud on renderdamisel näha), mille abil valmistada taimestikku ja/või ökosüsteeme, kasutades objekti, grupi või tahvli tüüpi visualiseerimist.

Välja kaalud (*Field Weights*)

Välja kaalude paneel laseb sul määrata, kui palju mõju igat tüüpi väline jõuväli või mõjutaja osakeste süsteemile avaldab. Jõuväljad on välised jõud, mis annavad dünaamilistele süsteemidele liikumise. Jõuväljade tüübid on detailsemalt kirjeldatud [jõuväljade peatükis](#).

Effector Group (mõjutajagrupp)

Limiteerib mõjutajad kindlale määratud grupile. Käesolevale süsteemile avaldavad mõju ainult mõjutajad selles grupis.

Gravity (gravitatsioon)

Määrab, kui palju mõju avaldab süsteemile globaalne gravitatsioon.

All (kõik)

Muudab kõigi mõjutajate kaalu suurust.

Jõuväljad (*Force Fields*)

Jõuvälja seadete paneel laseb sul panna iga üksiku osakese käituma nagu jõuvälja, lastes neil mõjutada teisi dünaamilisi süsteeme või isegi üksteist.

Self Effect (mõju iseendale)

Osakeste jõuväljad avaldavad mõju ka teistele osakestele samas süsteemis.

Amount (määr)

Määrab, kui paljud osakesed käituvad kui jõuväljad. 0 tähendab, et kõik nad on mõjutajad.

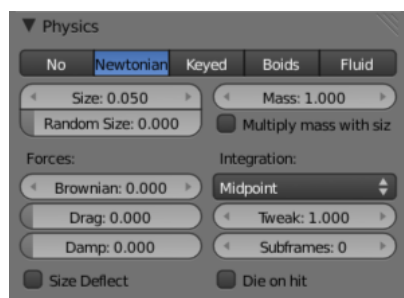
Osakeste süsteemile saab anda kuni 2 jõuvälja. Vaikimisi ei ole neil ühtegi. Nende sisse lülitamiseks vali rippmenüüst mõjutaja tüüp.

Seadeid on kirjeldatud [jõuväljade peatükis](#).

Newtoni füüsika

See on "normaalne" osakeste füüsika. Osakesed alustavad oma elu määratud algkiiruse ja nurkkiirusega ning liiguvad vastavalt Newtoni jõududele. Nende reageerimine keskkonna ja jõududega arvutatakse sõltuvalt animeerija poolt valitud integreerijast.

Jõud



Pilt 5: Newtoni füüsika

Brownian (Browni liikumine)

Määrab Browni liikumise määra. Browni liikumine lisab osakestele juhuslikku liikumist, mis põhineb Browni müraväljal. See on sobiv, et simuleerida nõrka puhanguelist tuult.

Drag (takistus)

Jõud, mis vähendab osakeste kiirust vastavalt nende kiirusele ja suurusele (kasulik õhutakistuse või veetakistuse simuleerimiseks).

Damp (summutamine)

Vähendab osakeste kiirust (aeglustus, hõõrdumine, summutamine).

Kokkupõrge

Size Deflect (suuruse järgi tõukamine)

Kasuta suunamisel osakeste suurust.

Die on Hit (sure tabamusel)

Tapab osakesed, kui need tabavad tõukavat objekti.

Integreerimine

Integreerijad on matemaatilised meetodid osakeste liikumise arvutamiseks. Järgnevad juhtnöörid aitavad valida sobiva integreerija, vastavalt animeerija poolt soovitud käitumisele.

Euler	Tuntud ka kui "ettepoole Euler". Kõige lihtsam integraator. Väga kiire, kuid mitte väga täpsete tulemustega. Kui summutamist ei kasutata, saavad osakesed aja jooksul aina rohkem energiat. Näiteks üksteise vastu pörkuvad osakesed pörkavad iga korraga aina kõrgemale ja kõrgemale. Mitte segi ajada "tagurpidi Euleri meetodiga" (seda ei ole sisse ehitatud), millel on vastupidine omadus: energia väheneb aja jooksul, isegi ilma summutamiseta. Kasuta seda integreerijat lühikeste simulatsioonide puhul koos ohtra summutamisega, kus kiired arvutused on olulisemad kui täpsus.
Varlet	Väga kiire ja stabiilne integreerija, energia säilib aja jooksul väga väheste kadudega.
Midpoint (keskpunkt)	Tuntud ka kui "2. astme Runge-Kutta". Aeglasem kui Euler, aga stabiilsem. Kui kiirendus on muutumatu (näiteks ei ole takistust), siis selle meetodi puhul energia säilib. Peab märkima, et pörkleivate osakeste näites võivad osakesed pörkata kõrgemale kui alguses, aga see ei ole pidev trend. Üldiselt on enamikul juhtudel hea kasutada seda integreerijat.
RK4	Lühend "4. astme Runge-Kutta" jaoks. Samane keskpunktile, kuid aeglasem ja enamasti ka täpsem. Süsteemi energia säilib, isegi kui kiirendus ei ole muutumatu. Seda on vaja ainult keerukates simulatsioonides, kus keskpunkt ei ole osutu küllalt täpseks.
Timestep (ajasamm)	Simulatsiooni ajasamm kaadri kohta.
Subframes (alamkaadrid)	Alamkaadrid, mida simuleerida, et parandada stabiilsust ja saavutada peenem teralisus. Kasuta kõrgemaid väärtusi kiiremini liikuvate osakeste puhul.

Võtmeosakesed



Pilt 6: Võtmeosakeste füüsika

Võtmeosakeste teekond on määratud emiteerijast teise osakeste süsteemi osakesteni. See võimaldab luua võtmeosakestest füüsika abil ahelaid ja nende abil pikki kiude või liikuvaid osakesi. Põhimõtteliselt ei ole nendel osakestel dünaamikat, kuid nad interpoleeritakse kindla aja jooksul ühest süsteemist teise. Kuna sul on selliste süsteemide üle väga täpne kontroll, saad sa neid kasutada

näiteks kiududega töötavate masinate jaoks (kangastelgede animatsioon vms). Pildil 3 voolavad kiud alumisest süsteemist (esimene võtmeosakeste süsteem) teise süsteemini keskel ja sealt ülemisse süsteemi, millel ei ole füüsikat. Kuna sul on võimalik vabalt animeerida igat emiteerivat objekti, saad sa teha ükskõik kui keerukaid animatsioone.

Ülesseadmine

Et valmistada ette võtmeosakesi, on sul vaja vähemalt kahte osakeste süsteemi.

Esimesel süsteemil on võtmeosakeste (Keyed) füüsika ja sel peab olema valik First (esimene) aktiveeritud. See hakkab olema nähtav süsteem. *Teine süsteem võib olla teine võtmeosakeste süsteem, kuid ilma valikuta First, või tavaline osakeste süsteem. Teine süsteem on juhitud süsteemi sihtmärgiks.

Loops (tsükleid)

Määrab, mitu korda osakeste liikumist läbi süsteemide korratakse. See on välja lülitatud, kui valik Use Timing (kasuta ajastust) on aktiveeritud.

Võtmed

Key Targets (võtmesihtmärgid)

Pead sisestama sihtsüsteemi oleva objekti nime, ja kui sellel on mitu osakeste süsteemi, siis ka süsteemi numbri.

Vajuta nuppu  võtme lisamiseks ja siis vali objekt.

Kui kasutad ainult ühte võtmesüsteemi, liiguvad osakesed oma eluaja jooksul emiteerijast sihtmärgini. Lühem eluiga tähendab kiiremat liikumist. Kui sul on rohkem kui üks võtmeosakeste süsteem järjest, jagatakse eluiga võrdselt nende vahel. Sel juhul võib osakeste kiirus sihtmärkide vahel varieeruda.

Ajastamine

Use Timing (kasuta ajastust)

Ajastamine toimib koos ahela teiste võtmeosakeste süsteemide liuguriga Time (aeg). Aja liugur laseb määrata osa osakese elueast tema liikumiseks.

Näiteks: oletame, et sul on ahelas kaks võtmeosakeste süsteemi ja kolmas süsteem sihtmärgiks. Esimese süsteemi osakeste eluiga on 50. Osakesed liiguvad 25 kaadri jooksul esimesest juhitud süsteemist teiseni ja järgmise 25 kaadri jooksul teisest süsteemist sihtmärgini. Kui kasutad ajastamisnuppu esimesel süsteemil, ilmub aja liugur teise süsteemi paneelile. Selle vaikeväärtus on 0.5, nii on aeg kahe süsteemi vahel võrdselt jaotatud. Kui sead aja (*Time*) väärtuseks 1, siis võtab liikumine esimesest süsteemist teise kogu eluea (osakesed surevad teises süsteemis).

Kui sead aja väärtuseks 0, siis alustavad osakesed liikumist teisest süsteemist ja liiguvad sihtmärgini.

Parved



Pilt 7: Parve füüsika

Parve (*Boid*) osakeste süsteeme saab panna teatud reeglite järgi käituma. Nad on kasulikud erinevate loomade, putukate ja kalade karjade, sülemite ja parvede simuleerimiseks. Nad on võimelised reageerima teiste objektide ja oma süsteemi liikmete asukohale. Parved suudavad töödelda ainult kindlat hulka informatsiooni, seetõttu on käitumise seadete järjekord väga oluline. Teatud kindlates situatsioonides kasutatakse ainult esimest kolme parameetrit.

Füüsika

Parved üritavad vältida objekte, millel on aktiveeritud suunamine (*Deflection*). Nad üritavad läheneda positiivse keraja väljaga (*Spherical Field*) objektidele ja eemalduda negatiivse keraja väljaga objektidest. Selle toimimiseks peavad objektid peavad jagama vähemalt ühte ühist kihti. Seda ühist kihti ei ole vaja renderdada, seega saad kasutada nähtamatuid mõjutajaid.

Parved võivad füüsikat muuta vastavalt sellele, kas nad asuvad õhus või maapinnal (kokkupõrke objektile)

Allow Flight (luba lennata)

Lubab parvedel liikuda õhus.

Allow Land (luba maapinnal)

Lubab parvedel liikuda maapinnal.

Allow Climbing (luba ronida)

Lubab parvedel ronida eesmärgiks olevatele objektidele.

Max Air Speed (maksimaalne kiirus õhus)

Määrab maksimumkiiruse õhus.

Min Air Speed (minimaalne kiirus õhus)

Määrab miinimumkiiruse õhus.

Max Air Acceleration (maksimaalne kiirendus õhus)

Külgkiirendus õhus, protsent maksimumkiirusest (pööre). Määrab, kui kiiresti parv on võimeline suunda muutma.

Max Air Angular Velocity (maksimaalne nurkkiirus õhus)

Radiaalkiirendus õhus, protsent 180st kraadist. Määrab, kui palju parv on võimeline äkitselt kiirendama, et reeglilt järgida.

Air Personal Space (isiklik ruum õhus)

Parve isikliku ruumi raadius õhus. Protsent osakese suurusest.

Landing Smoothness (maandumise sujuvus)

Kui sujuvalt parved maanduvad.

Max Land Speed (maksimaalne kiirus maapinnal)

Määrab maksimumkiiruse maapinnal.

Jump Speed (hüppe kiirus)

Maksimaalne hüppekiirus

Max Land Acceleration (maksimaalne kiirendus maapinnal)

Külgkiirendus maapinnal, protsent maksimumkiirusest (pööre). Määrab, kui kiiresti parv on võimeline suunda muutma.

Max Land Angular Velocity (maksimaalne nurkkiirus maapinnal)

Radiaalkiirendus maapinnal, protsent 180st kraadist. Määrab, kui palju parv on võimeline äkitselt kiirendama, et reeglilt järgida.

Land Personal Space (isiklik ruum maapinnal)

Parve isikliku ruumi raadius maapinnal. Protsent osakese suurusest.

Land Stick Force (maapinna kleepjõud)

Kui tugevat jõudu on vaja, et hakata mõjutama parve maapinnal.

Banking (kaldumine)

Ümber kiirusvektori pöördumise suurus pöoretel. Kalle (1.0 == loomulik kalle).

Pitch (üles/alla pööre)

Pöörde suurus ümber külgvektori.

Height (kõrgus)

Parve kõrgus võrreldes osakeste suurusega.

Võitlus

Health (tervis)

Parve algne tervis sündides.

Strength (tugevus)

Maksimaalne kahju rünnakul sekundis.

Aggression (agressioon)

Parv võitleb niimitu korda tugevamini kui vastane.

Accuracy (täpsus)

Rünnaku täpsus.

Range (vahemaa)

Maksimaalne parve rünnakukaugus.

Liit (*Alliance*)

Suhete (*Relations*) kast lubab sul panna teisi osakeste süsteeme reageerima parvedega. Määrates tüübiks Enemy (vaenlane) paneb see süsteemid omavahel võitlema. Friend (sõber) paneb süsteemid koostööd tegema. Neutral (neutraalne) ei pane neid ei omavahel liitu moodustama ega võitlema

Suunajad ja mõjutajad (*Deflectors and Effectors*)

Nagu juba varem mainitud, reageerivad parved, täpselt nagu Newtoni osakesed, ümbritsevatele suunajatele ja väljadele vastavalt animeerija vajadustele:

Suunamine (*Deflection*): Parved üritavad vältida mõjutavat objekti vastavalt põrkereegli (*Collision*) kaalule. See toimib kõige paremini kumerate pindade puhul (nõgusad pinnad vajavad veel tööd). Parve füüsika jaoks määravad objektide väljanägemise kerajad väljad. Nii on negatiivne kerajas väli (või objekt või osakeste süsteem) kõigile teistele parvedele kiskjaks ja positiivne väli eesmärgiks.

Kui valid objekti, millele on seatud osakeste süsteem, on sul väljade (*Fields*) kaardil väike menüü, mis määrab, kas välja kohaldatakse emiteerivate objektile või osakeste süsteemile. Kui soovid, et saakosakesed eemalduks kiskjaosakestest, pead sa määrama osakeste süsteemi nime.

Kerajad väljad (*Spherical fields*): Need mõjutajad võivad olla kiskjad (negatiivse jõuga), mida parved üritavad vältida, või sihtmärgid (positiivse jõuga), mille poole nad püüdlevad vastavalt vältimise (*Avoid*) ja eesmärgi (*Goal*) reeglite kaaludele. Kerajate väljade tegelikult mõjuv jõud korrutatakse vastava tegeliku kaaluga (näiteks, kui kas tugevuse või eesmärk on null, siis parved ei hakka sellele positiivsele väljale järgnema). Samuti saad sa aktiveerida "suma tabamisel" (*Die on Hit*) (lisade (*Extras*) paneelil) nii, et saakosakesed kaovad, kui nendeni jõudnud kiskjaosakesed on neid "rünnanud". Et see toimiks, peab kiskjaosakestel olema negatiivse jõuga kerajas väli (kuid sa võid selle tugevuseks seada -0.01) – ei piisa sellest, kui saakosakestele määrata positiivse jõuga eesmärk. Kiskja- ja saakosakeste suuruse saab määrata nupuga *Size* (suurus) paneelil *Extras* (lisad).

Parve aju (*Boid Brain*)

Boid Brain (parve aju) paneel määrab, kuidas parve osakesed reageerivad üksteisele. Parve käitumine määratakse rea reeglitega. Nimekirjast on võimalik arvesse võtta ainult kindlat hulka informatsiooni. Kui mälumaht ületatakse, ignoreeritakse ülejäänud reegleid.

Vaikimisi arvestatakse reegleid nimekirjas ülevalt alla (andes neile selged prioriteedid), nende järjekorda saab muuta, kasutades paremal pool asuvaid väikseid nooli.

Võimalik reeglite nimekiri:

Goal (eesmärk)

Otsi eesmärki (objekte keraja välja ja positiivse jõuga)

Predict (ennusta)

Ennusta sihtmärgi liikumist

Avoid (väldi)

Väldi "kiskjaid" (objekte keraja välja ja negatiivse jõuga)

Predict (ennusta)

Ennusta sihtmärgi liikumist

Fear Factor (hirmufaktor)

Väldi objekti, kui sellest lähtuv oht on määratust suurem

Avoid Collision (väldi kokkupõrkeid)

Väldi objekte, millel on aktiveeritud suunamine (deflection)

Boids (parved)

Väldi põrget teiste parvedega

Deflectors (suunajad)

Väldi põrget suunavate objektidega

Look Ahead (vaata ette)

Aeg sekundites, mille ulatuses ettepoole vaadatakse

Separate (eralda)

Parved liiguvad üksteisest eemale

Flock (koos liikumine)

Kopeerivad naabruses asuvate parvede liikumist, vältides samal ajal üksteist

Follow Leader (järgi juhti)

Järgib parve asemel liiderobjekti

Distance (vahemaa)

Vahemaa, mis kaugusel liidrile järgneda

Line (joon)

Järgne liidrile reas

Average Speed (keskmine kiirus)

Säilita keskmine kiirus.

Speed (kiirus)

Protsent maksimumkiirusest

Wander (uita)

Liikumise suuna juhusliku muutumise kiirus

Level (tase)

Kui muutumatuna hoitakse liikumise Z osa

Fight (võitle)

Liigu lähedalasuvate parvede poole

Fight Distance (võitluse kaugus)

Ründa parvi maksimaalselt sellelt kauguselt

Flee Distance (põgenemise kaugus)

Põgene sellele kaugusele

Reeglite arvestamine

Reeglite arvestamise määramiseks on kolm meetodit.

Average (keskmine)

Kõik reeglid on võrdsustatud.

Random (juhuslik)

Igale parvele valitakse juhuslik reegel.

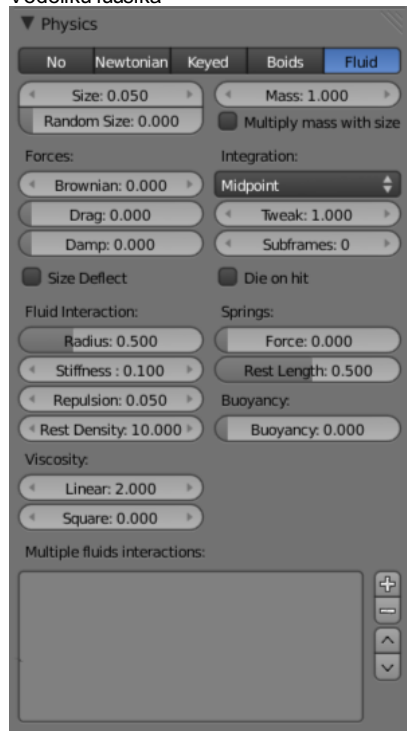
Fuzzy (hägune)

Reegleid arvestatakse ähmase loogika (*fuzzy logic*) järgi. Reeglid võetakse täitmiseks ülevalt alla. Arvestatakse ainult esimest reeglit, mille mõju on suurem ähmasuse lävest. Väärtus näitab ära, kui palju parv üritab antud reeglit täita (väärtusel 1.000 teeb ta seda alati, väärtusel 0.000 mitte kunagi). Kui parvel on samal ajal rohkem kui üks konfliktis olev reegel, proovib ta täita neid kõiki vastavalt nende väärtusele.

Pane tähele, et antud parv teeb kõik mis võimalik, et täita kõiki talle antud reegleid, kuid on tõenäoline, et mõnedel juhtudel saab mõni reegel eelise. Näiteks, et vältida kiskjat, võib parv "unustada" põrke (*Collision*), karja (*Crowd*) ja keskpunkti (*center*) reeglid, mis tähendab, et olles "paanikas", võib ta pörgata vastu takistusi, isegi kui talle on määratud seda mitte teha.

Lõpetuseks - põrke algoritm ei ole siiani täiuslik ja on arendamisel, nii et mõnedel juhtudel on oodata vale käitumist. Selle kallal töötatakse edasi.

Vedeliku füüsika



Pilt 8: Vedeliku füüsika

Vedelikusimulatsioone kasutatakse arvutigraafikas laialdaselt ja nad on iga osakeste süsteemi väga nõutud omadus. Vedelikuosakesed on samased Newtoni osakestele, aga siin mõjutavad osakesi sisemised jõud nagu surve, pindpinevus, viskoossus, vetruvus jne. Blenderi vedelikuosakesed kasutavad vedelikuvõrrandite lahendamiseks SPH tehnikat.

Silutud osakeste hüdrodünaamika (*Smoothed-particle hydrodynamics*, SPH) on arvutuslik meetod, mida kasutatakse vedeliku voolamise simuleerimiseks. Seda kasutatakse mitmetes teadusharudes, kaasa arvatud astrofüüsikas, ballistikas, vulkanoloogias ja okeanograafias. See on võrevaba Lagrange'i meetod (kus koordinaadid liiguvad koos vedelikuga) ja meetodi lahutust saab lihtsalt korrigeerida muutujate, näiteks tiheduse abil.

Vedelikust limani, vahust liiva ja tiheda suitsuni, võimalused on lõputud.

Seaded

Vedeliku füüsika jagab [Newtoni füüsika](#) seadeid. Neid vaadeldakse vastaval lehel.

Vedeliku omadused

Stiffness (jäikus)

Kui kokkusurumatu vedelik on.

Viscosity (viskoossus)

Lineaarse viskoossus. Paksema vedeliku jaoks kasuta madalamat viskoossuse väärtust.

Buoyancy (ujuvus)

Kunstlik ujuvusjõud negatiivses gravitatsiooni suunas, mis põhineb vedelikusisesel rõhuerinevusel.

Keerulisemad

Repulsion Factor (tagasitõuke tegur)

Kui tugevalt vedelik üritab vältida kobardumist (suhe jäikusesse). Märkeruut määrab tagasitõukejõu suhtena jäikusesse.

Stiff Viscosity (viskoossus)

Tekitab viskoossuse laialivoolavale vedelikule. Märkeruut määrab selle suhtena normaalsesse viskoossusesse.

Interaction Radius (vastasmõju raadius)

Vedeliku mõjuraadius. Märkeruut määrab selle suhtena neljakordsesse osakese suurusesse.

Rest Density (puhkeoleku tihedus)

Paigaloleva vedeliku tihedus. Märkeruut määrab selle suhtena vaiketihedusse.

Vedrud

Force (jõud)

Vetrumisjõud

Rest Length (puhkeoleku pikkus)

Pikkus puhkeasendis. Suhe osakeste raadiusesse. Märkeruut määrab selleks suhtena kahekordsesse osakese suurusesse.

Viscoelastic Springs (viskoos-elastsed vedrud)

Kasuta viskoos-elastseid vedrusid Hooke'i vedrude asemel.

Elastic Limit (elastsuspiir)

Kui palju peab vedru kokku pressima/venitama, et tema algpikkus muutuks

Plasticity (plastilisus)

Kui palju algpikkus muutub, kui elastsuslimiit on ületatud.

Initial Rest Length (algne lähtepikkus)

Kasuta algset pikkust vedru puhkeoleku pikkusena kahekordse osakese suuruse asemel.

Frames (kaadreid)

Loob vedrud osakese sünnist (see on alati 0) kuni niimitu kaadrit edasi.

Osakeste visualiseerimine

Esituse (Display) ja renderdamise (Render) paneelil saad sa määrata erinevaid viise, kuidas osakesi renderdatakse või 3D-vaates näidatakse. Mõned valikud kehtivad ainult 3D-aknas, osakesed renderdatakse sel juhul alati nagu [halod](#). Mõned valikud renderdatakse, nagu 3D-aknas näha on.

Esitus 3D-vaates (*Viewport Display*)

Esituse (*Display*) paneel määrab, mismoodi osakesi 3D-vaates kuvatakse. See ei määra tingimata, mismoodi nad renderdatuna paistavad.

None (puudub)

Osakesi ei ole 3D-aknas näha ja neid ei renderdata. Kuigi emiteerija võib olla renderdatud.

Point (punkt)

Osakesi näidatakse kandiliste punktikestena. Nende suurus on kaamera kaugusest sõltumatu.

Circle (ring)

Osakesi näidatakse vaate suunas olevate ringidena. Nende suurus on kaamera kaugusest sõltumatu.

Cross (rist)

Osakesi näidatakse 6-haruliste tärnidena, mille paigutus vastab osakeste pöördede. Nende suurus on kaamera kaugusest sõltumatu.

Axis (telg)

Osakesi näidatakse 3-haruliste telgedena. See on kasulik, kui soovid vaateavas näha osakeste orientatsiooni ja pöörlemist. Suurenda seadet Draw Size (joonistuse suurus), kuni eristad telgesid selgesti.

Osakestel, mis on visualiseeritud punkti, ringi, täрни või teljena, ei ole mingeid erilisi valikuid - kuid need võivad osutada väga kasulikuks, kui sul on tegu mitme osakeste süsteemiga ja sa ei taha ühe süsteemi osakesi teistega segamini ajada (näiteks simulatsioon, mis kasutab parvede (*Boids*) füüsikat).

Display (esitus)

Määrab protsentuaalselt, kui palju osakesi vaateavas näidatakse (renderdatakse ikkagi kõik osakesed).

Draw Size (joonistuse suurus)

Määrab, kui suurelt (pikslites) osakesi vaateavas näidatakse (0 = vaikeväärtus).

Size (suurus)

Näitab osakeste suurust ringina.

Velocity (kiirus)

Näitab osakeste liikumist joonega, mis osutab liikumise suunda ja mille pikkus on võrdeline kiirusega.

Number (number)

Näitab osakeste id-numbreid emissiooni järjekorras.

Värv (*Color*)

Värvi menüü lubab sul muuta osakeste esitust vastavalt nende teatud omadustele.

None (puudub)

Osakesed on mustad.

Material (materjal)

Osakesed on värvitud vastavalt neile antud materjalile.

Velocity (kiirus)

Osakesed on värvitud vastavalt nende kiirusele. Värvüleminek on sinisest roheliseni ja edasi punaseni, sinine on aeglaseim ja punane on kiirus, mis läheneb välja Max (maksimaalse) väärtuseni või üle selle. Suurendades välja Max (maksimaalne) väärtust, on võimalik saavutada laiem osakeste kiiruste vahemik.

Acceleration (kiirendus)

Osakesed on värvitud vastavalt nende kiirendusele.

Renderdamise seaded (*Render Settings*)

Renderduspaneel määrab, mismoodi osakesed paistavad renderdatuna.

Material Index (materjali indeks)

Määrab, millist objekti materjalidest kasutatakse osakeste varjutamiseks.

Parent (ülem)

Kasutab erineva objekti koordinaate, et määrata osakeste sünd.

Emitter (emiteerija)

Kui see on välja lülitatud, siis emiteerijat ei renderdata. Aktiveeri nupp Emitter (emiteerija), et renderdada ka võre.

Parents (ülem)

Kui kasutatakse alamosakesi, renderdab ka ülemosakesed. Alamosakestel on palju erinevaid deformatsiooni seadeid ja sirged ülemosakesed torkaks nende vahel silma. Seega kui alamosakesed (Children) on aktiveeritud, siis ülemosakesi (Parents) vaikimisi ei renderdata. Vaata [Alamosakesed](#)

Unborn (sündimata)

Renderdab osakesed, enne kui nad sünnivad.

Died (surnud)

Renderdab osakesed pärast nende surma. See on väga kasulik, kui osakesed surevad pörkel (Die on hit), nii saad sa objekte osakestega katta.

Puudub (*None*)

Kui väärtuseks on määratud None (puudub), siis osakesi ei renderdata. See on kasulik, kui kasutad osakesi objektide duplitseerimiseks.

Halo

Halo osakesed renderdatakse [halo tüüpi materjalina](#).

Trail Count (jälgede arv)

Määrab järgivate osakeste arvu. Kui see on suurem kui 1, näidatakse lisavalikuid.

Length in Frames (pikkus kaadrites)

Trajektoori ajastus absoluutsetes kaadrites.

Length (pikkus)

Joonistatud trajektoori lõpuaeg.

Random (juhuslik)

Annab osakeste trajektoori pikkusele juhuslikud variatsioonid.

Joon (Line)

Joone visualiseerimisrežiim loob kiu renderdaja abil (peenemad või jämedamad) segmentidest koosnevad jooned osakeste liikumise suunas. Joone paksus määratakse parameetriga Start (algus) kiudude varjutaja (Strands shader) materjalide alamkonteksti (Material sub-context) seoste ja protsesside (Links and Pipeline) paneelil.

Back (taga)

Määrab osakeste saba pikkuse.

Front (ees)

Määrab osakeste pea pikkuse.

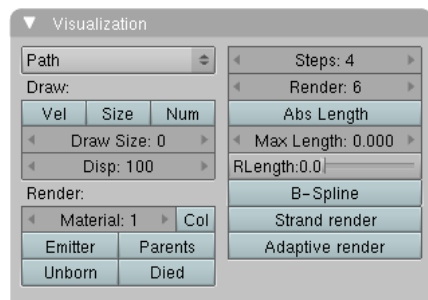
Speed (kiirus)

Korrutab joone pikkuse osakese kiirusega. Mida kiirem, seda pikem joon.

Trail Count (jälgede hulk)

Vaata [halo renderdustüübi](#) kirjeldust ülalpool.

Trajektoor (Path)



Pilt 3: Visualisatsiooni paneel trajektoori visualiseerimiseks

Trajektoori (Path) visualiseerimine vajab [karva](#) (Hair) osakeste süsteemi või [võtmeosakesi](#) (Keyed Particles).

Strand render (kiu renderdaja)

Kasuta renderdamisel kiu primitiive. Väga kiire ja efektiivne renderdaja.

Adaptive render (adaptiivne renderdaja)

enne osakese kiudude renderdamist üritab eemaldada üleliigse geomeetria, et muuta renderdamist kiiremaks ja mälule lihtsamaks.

Angle (nurk)

Kui mitu kraadi tee peab kõverduma, et tekitada järgmine renderdussegment (tee sirged osad vajavad vähem segmente).

Pixel (piksel)

Kui palju piksleid peab tee läbima, et tekitataks järgmine renderdussegment (väga lühikesed karvad või pikad karvad kaugelt vaadatuna vajavad vähem segmente). (Ainult adaptiivsel renderdamisel.)

B-Spline (B-splain)

Interpoleeri karvad, kasutades B-splaini. See võiks olla sobiv valik, kui soovid kasutada madalaid renderduse väärtusi. Sa kaotad küll natuke juhtimisvõimalusi, kuid saad sujuvamad trajektoorid.

Steps (samme)

Määrab renderdatud trajektooride tükeldamise arvu (väärtus on 2 astmel n). Seda väärtust peaks seadma ettevaatlikult, sest kui suurendad väärtust kahe võrra, vajad renderdamiseks neli korda rohkem mälu. Samuti on renderdamine kiirem, kui kasutad madalaid renderduse väärtusi (mõnikord drastiliselt kiirem). See, kui madalaks saad selle väärtusega minna, sõltub sellest, kui lainelised karvad on (väärtus on 2 astmel n). See tähendab, et 0 sammu annab 1 tükelduse, 1 annab 2 tükeldust, 2→4, 3→8, 4→16, ... $n \rightarrow 2^n$.

Ajastamise valikud:

Absolute Path Time (absoluutne trajektoori aeg)

Trajektoori ajastamine absoluutkaadrites.

Start (algus)

Joonistatud trajektoori algusaeg.

End (lõpp)

Joonistatud trajektoori lõpuaeg.

Random (juhuslik)

Annab trajektoori pikkusele juhuslikud variatsioonid.

Põhjalikumaks kirjelduseks vaata käsiraamatu peatükki [Karvad](#).

Objekt (*Object*)

Objekti visualiseerimisrežiimis duplitseeritakse määratud objekt (väli Dupli Object - duplitseeritav objekt) iga osakese asemele. Duplitseeritud objekt peab olema koordinaatsüsteemi keskel, vastasel juhul nihutab ta osakesi.

Global (globaalne)

Kasutab duplitseerimiseks objekti globaalseid koordinaate.

Size (suurus)

Objektide suurus

Random Size (juhuslik suurus)

Annab objektide suurustele juhuslikud variatsioonid.



Grupp (*Group*)

Grupi visualiseerimisrežiimis duplitseeritakse gruppi kuuluvad objektid (väli GR:) järjest osakeste asemele.

WholeGroup (terve grupp)

Kasutab tervet gruppi korraga selle elementide asemel, iga osakese asemele kuvatakse grupp.

Use Count (kasutuskordi)

Kasutab samas grupis objekte mitu korda. Määra ilmuvas nimekirjas iga objekti jaoks, mitu korda ja mis järjekorras teda korratakse. Sa saad duplitseerida objekti nimekirjas nupuga  ja eemaldab duplikaadi nupuga .

Use Global (kasuta globaalset)

Kasutab duplitseerimiseks objekti globaalseid koordinaate.

Pick Random (vali juhuslik)

Grupis olevad objektid valitakse juhuslikus järjekorras ja iga osakese asemel kuvatakse ainult ühte objekti.

Pane tähele, et see mehhanism asendab täielikult Blenderi vana osakeste süsteemi, mis kasutas ülem-alam süsteemi ja tippude duplitseerimist osakeste geomeetriaga asendamiseks. See meetod on täielikult iganenud ja ei toimi enam.

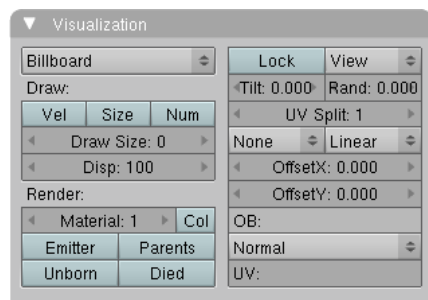
Size (suurus)

Objektide suurus

Random Size (juhuslik suurus)

Annab objektide suurustele juhuslikud variatsioonid.

Tahvliid (*Billboard*)



Pilt 4: Osakeste tahvli (Billboard) visualiseerimisrežiim

Tahvliid (Billboards) on joondatud kandilised tasapinnad. Vaikimisi on nad joondatud kaamera järgi, aga sa saad valida teise objekti, mille järgi neid joondada.

Kui liigutad tahvli ümber tema sihtobjekti, on ta alati küljega sihtobjekti keskkoha suunas. Tahvli suurus määratakse osakese suuruse (Size) parameetriga (Blenderi ühikutes). Sa saad neid kasutada näiteks [spraitide](#) jaoks või halo visualiseerimise asendamiseks. Kõike, mida saab teha haloga, saab teha ka tahvlitega. Aga tahvliid on reaalsed objektid, neid näeb kiirtejälitus, nad paistavad läbipaistvate objektide tagant, neil võib olla suvaline vorm ja nad võtavad vastu valgust ja varje. Neid on natuke raskem luua ja nad võtavad rohkem renderdusaega ja ressursse.

Tahvlite tekstuurimine (kaasa arvatud animeeritud tekstuurid ja katvus) tehakse, kasutades UV-koordinaate, mis genereeritakse neile automaatselt - seega saavad nad olla suvalise kujuga. See toimib hästi animatsioonide puhul, sest tahvlite joondamine võib olla dünaamiline. Tekstuure võib animeerida erinevate meetoditega:

- Olenevalt osakese elueast (suhteline aeg).
- Olenevalt osakeste algusajast.
- Olenevalt kaadrist (absoluutne aeg).

Sa võid kasutada pilditekstuuri erinevaid osi:

- Olenevalt tahvli elueast.
- Olenevalt emissioonijast.
- Olenevalt joondamisest või kaldest.

Kuni kasutad tahvli jaoks harilikke materjale, on sul täielik vabadus segada tekstuure oma äranägemise järgi. Materjal ise animeeritakse absoluutajas.

Kõige olulisem on aru saada, et kui objektile ei ole ühtegi UV-kihti (*UV Layer*), pead sa muutmiskontekstis looma vähemalt ühe, et mõni nendest meetoditest töötaks. Lisaks tuleb tekstuurile määrata laotus UV-koordinaatide järgi paneelil Map Input (sisendi seostamine). Kui soovid näha mõningaid näiteid sellise animatsiooni võimalustest, vaata [Tahvlite animatsiooni õppetükki](#).

Mõningatel juhtudel on huvitavaks alternatiiviks tahvlitele karvad, sest sa saad animeerida karvade kuju. Kuna sellel visualisatsioonitüübil on nii palju valikuid, selgitatakse neid detailsemalt allpool.

Nende valikutega saad sa liikumist piirata. Kuidas telg on emissiooni ajal eeljoondatud.

View (vaade)

Eeljoondust ei ole, tavaline orientatsioon sihtobjektile.

X/Y/Z

Vastavalt mööda globaalset X/Y/Z-telge.

Velocity (kiirus)

Piki osakese kiirusvektorit.

Lock (lukusta)

Lukustab joondamistelje ja hoiab seda orientatsiooni, tahvel joondub sihtobjektile ainult ühte telge mööda.

Billboard Object (tahvli objekt)

Sihtobjekt, mille poole tahvel küljega on. Vaikimisi kasutatakse aktiivset kaamerat.

Tilt Angle (kaldenurk)

Tahvliatasapindade pöördnurk. Kalde väärtus 1 pöörab tahvli 180 kraadi (keerab tahvli teistpidi).

Random (juhuslik)

Juhuslik kalde varieerimine.

Offset X (X nihe)

Nihutab tahvli horisontaalselt osakese keskpunkti suhtes, see ei liiguta tekstuuri.

Offset Y (Y nihe)

Nihutab tahvli vertikaalselt osakese keskpunkti suhtes.

UV Channels (UV-kanalid)

Tahviid on lihtsalt neljakandilised hulknurgad. Et neid eri viisidel tekstuuri, peab meil olema meetod, mis määraks, mis tekstuure me tahvli jaoks soovime ja kuidas me tahame, et nad oleks nelinurkadele laotatud. Seda saab määrata tekstuuri laotamise nuppudega, sättides soovitud tekstuuri erinevatele koordinaatidele. Sa võid kasutada kolme erinevat UV-kihti ja saada kolm erinevat komplekti UV-koordinaate, mida saab siis rakendada erinevatele (või samale) tekstuurile.

Billboard Normal UV (tahvli normaalne UV)

Koordinaadid on igal tahvil samad, aseta lihtsalt pilt otse nelinurgale.

Billboard Time-Index (X-Y) (tahvli aja-indeksi UV)

Koordinaadid määravad tegelikult üksikud punktid tekstuuri, kus X-telg on aeg ja Y-telg on osakese number. Näiteks kui tekstuuri on horisontaalne üleminek valgest mustani ja seostatud värviga, tekitab see meile osakesed, mis alustavad valgena ja muutuvad oma eluea jooksul vähehaaval mustaks. Teisest küljest tekitab vertikaalne üleminek valgest mustani esimese osakese valgena ja viimase mustana, osakesed nende vahel on erinevat halli värvi.

UV-tekstuuride animeerimine on natuke keeruline. UV-tekstuuri on jagatud ridadeks ja veergudeks (N korda N). Tekstuuri peaks olema ruudukujuline. Sa pead kasutama seadet UV Split UV-kanalil ja andma UV-kihile nime. See tekitab sellele kihile UV-koordinaadid.

Split UV's (jagatud UV-koordinaadid)

Tekstuuri kasutatavate ridade/veergude arv.

Koordinaadid on üks osa jagatud UV (UV Split) ruudustikust, mis on $n \times n$ ruudustik üle kogu tekstuuri. Millist tekstuuri osa millisel osakesel ja mis ajal kasutatakse, on määratud väljade Offset (nihe) ja Animate (animeeri) väärtustega. Neid võib kasutada, et teha iga tahvel erinevaks, või kasutada nendel "animeeritud" tekstuuri, mille iga kaader on erineval suure pildi ruudul.

Billboard Split UV (tahvli jagatud UV)

Tahvil kasutatava UV-kihi nimi (sa võid kasutada iga UV-kanali jaoks erinevat). Vaikimisi on selleks aktiivne UV-kiht (kontrolli seda muutmiskonteksti Edit võre paneelil Mesh, kiirklahv F9).

Animate (animeeri)

Rippmenüü, mis määrab, kuidas jagatud UV-koordinaate peaks animeerima (muutus osakesest osakeseni läbi aja):

None (puudub)

Osakesel endal ei ole animatsiooni, tahvel kasutab kogu eluaja jooksul tekstuuri ühte osa.

Age (iga)

Tekstuuri sektsioone kasutatakse osakese eluea jooksul järgemööda.

Angle (nurk)

Muudab sektsiooni vastavalt pöördle ümber telje, mille järgi ta on joondatud (Align to). Kui kasutatakse seadet View (vaade), siis sõltub muutus kaldest.

Frame (kaader)

Sektsiooni muudetakse vastavalt kaadriks.

Offset (nihe)

Määrab, kuidas valitakse esimene sektsioon (kõigist tekstuuri osadest $n \times n$ ruudustikul, mis on määratud seade UV Split numbriga) kõikide osakeste jaoks.

None (puudub)

Kõik osakesed algavad esimesest sektsioonist.

Linear (lineaarne)

Esimene osake algab esimesest sektsioonist ja viimane osake algab viimasest sektsioonist, osakesed nende vahel algavad sektsioonist, mis määratakse järjest esimesest viimaseni.

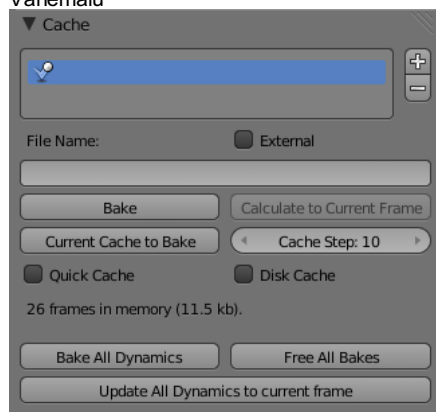
Random (juhuslik)

Annab alguses igale osakesele juhusliku sektsiooni.

Trail Count (jälgede arv)

Vaata [halo renderdustüübi](#) kirjeldust ülalpool.

Vahemälu



Pilt 4: Osakeste vahemälu (Cache) paneel.

Emiteerija (Emitter) tüüpi süsteemid kasutavad ühtset vahemälu kasutamise ja eeltöötluse süsteemi (koos pehmete kehade ja riidega). Simulatsiooni tulemused salvestatakse animatsiooni esitamisel automaatselt kettale, nii et järgmine kord seda käivitades loeb ta tulemusi kettalt ja mängib kiiremini. Kui sa eelarvutad (Bake) simulatsiooni, siis on vahemälu kaitstud ja sult küsitakse kinnitust, kui proovid muuta seadeid, mis teevad vajalikuks ümberarvutamise.



Ettevaatust alguse (Start) ja lõpu (End) seadetega

Simulatsioon arvutatakse ainult eeltöötluse (Bake) paneeli alguse (Start) ja lõpu (End) kaadrite vahemikus, olenemata sellest, kas sa kasutad eeltöötlust või mitte. Seega, kui tahad, et simulatsioon oleks pikem kui 250 kaadrit, pead sa muutma kaadri End (lõpp) väärtust!

Vahemälu kasutamine

- Kui animatsiooni esitatakse, kirjutab iga füüsikasüsteem kõik algus- ja lõppkaadri vahelised kaadrid kettale. Need failid salvestatakse .blend-failiga koos alamkaustadesse, mille nimed algavad eesliitega "blendcache". NB! Vahemälu täitmiseks peab taasesitus algama enne, või samast kaadrist, kust simulatsioon algab.
- Vahemälu tühjendatakse automaatselt, kui teed muudatusi – aga mitte alati, mõnikord on vaja seda käsitsi tühjendada. Näiteks kui oled teinud muudatusi jõuväljades.
- Vahemälu ei kasutata, kui alamkaustadesse kirjutamine ei ole võimalik.
- Vahemälu saab vabastada iga füüsikasüsteemi kohta eraldi paneelil oleva nupuga või siis kiirklahviga CtrlB kõigil valitud objektidel korraga.
- Kui faili täieliku asukoha nimi vahemälus on pikem kui operatsioonisüsteem võimaldab (näiteks üle 250 tähemärgi), võib kummalisi asju juhtuda.

Eeltöötlus

- Pärast eeltöötlust on süsteem muudatuste eest kaitstud.
- Eeltöötluse (Bake) tulemused tühistatakse kõigi valitud objektide puhul kiirklahviga CtrlB või klõpsates nupule Free Bake (tühista eeltöötlus) üksiku osakeste süsteemi jaoks.
- Kui võret muudetakse, ei arvutata simulatsiooni uuesti.
- Kahjuks ei ole eeltöötlust pehmetele kehadele ja riidele.

Lõpetuseks kaks märkust:

- Renderfarmide puhul on kõige parem kõik füüsikasüsteemid eeltöödelda ja kopeerida blendcache-kaust samuti renderfarmi.
- Nagu alati, ole tähelepanelik töötlejate järjestusega pinus. Sul võib mõnikord olla 3D-vaatel ja renderdamisel erinev arv külgi (näiteks kui kasutad pinnatükeldamist); kui nii, siis võib renderdatud tulemus olla suuresti erinev sellest, mida sa näed 3D-vaates.

Karvad

Karvade režiimis loob osakeste süsteem ainult staatilisi osakesi, mida saab kasutada juuste, karvade, muru ja muu taolise tegemiseks.

Kasvamine

Esimene samm on luua karvad, määrates nende kiudude arvu ja pikkuse.

Karva jaoks arvutatakse korraga välja terve osakeste liikumistee. Seega saavad karvad teha kõike, mida teeb osake. Karva pikkus on sama, nagu oleks 100-kaadrilise elueaga osakese liikumisteel. Selle asemel, et renderdada osakese animatsiooni punkt-punkti haaval igas kaadris eraldi, arvutatakse välja interpolateeritavad kontrollpunktid, segmendid.

Kujundamine

Järgmine samm on karvade kujundamine. Sa võid muuta põhikarvade välimust, muutes [füüsika seadeid](#).

Täiuslikum vahend karvade välimuse muutmiseks on [alamosakeste](#) kasutamine. See lisab juurde esialgsetest karvadest sõltuvaid karvu ja siin on seaded neile erinevat tüüpi vormide andmiseks.

Samuti saad interaktiivselt karvu kujundada [osakeste režiimis](#). Selles režiimis muutuvad osakeste seaded mitteaktiivseks ja sa saad karva kõveraid kombineerida, trimmida, pikendada jne.

Animeerimine

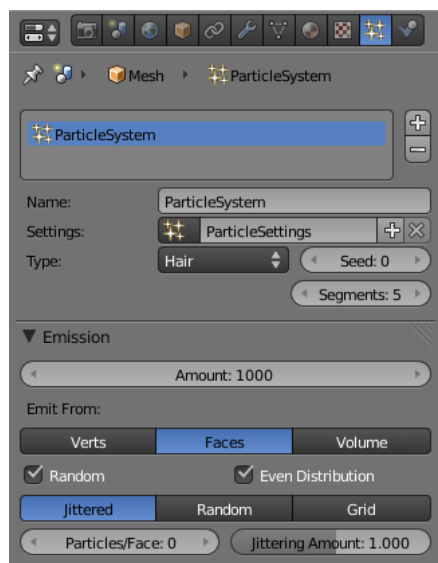
Kasutades riidesimulaatorit, saab nüüd karvad muuta dünaamiliseks. Seda käsitletakse [karvade dünaamika](#) leheküljel.

Renderdamine

Blender saab karvu renderdada mitmel erineval viisil. Materialidel on kiudude seaded, mis on käsitletud materialide peatüki [kiudude](#) leheküljel.

Karvad võivad olla ka aluseks [osakesetöötlejale](#), mis lubab võret deformeerida piki kõveraid, mis on kasulik jämedamate kiudude, muru või sulgede jaoks, millel võib olla spetsiifilisem väljanägemine.

Valikud



Pilt 4a: Karvade (Hair) osakeste süsteemi seaded.

Regrow (kasvata uuesti)

Kasvatab karva uuesti iga kaadri jaoks.

Advanced (keerulisemad)

Lülitab sisse keerulisemad seaded, mis vastavad samadele seadetele nagu töötades emiteerija režiimis.

Emissioon

Amount (hulk)

Määrab kiudude hulga. Kasuta nii väheseid osakesi kui võimalik, eriti kui plaanid hiljem kasutada pehme keha animatsiooni. Aga osakesi peab olema piisavalt palju, et neid saaks täpselt juhtida. "Normaalse" soengu puhul annavad küllaldase kontrolli minu kogemuse järgi mõnituhat (umbes 2000) osakest. Kui plaanid kogu keha karvadega katta, vajad sa palju rohkem osakesi. Maht tekitatakse hiljem alamosakeste (Children) abil.

Karvade dünaamika

Karvadele liikumise lisamiseks vaata [karvade dünaamika](#) peatükki.

Esitus

Rendered (renderdatud)

Joonistab karvad kõveratena.

Path (rada)

Joonistab ainult karvade otsad.

Steps (samme)

Karva kiudude segmentide arv (kontrollpunktide arv miinus 1) . Kontrollpunktide vahel segmentide väärtused interpoleeritakse. Kontrollpunktide arv on oluline:

- seetõttu, et pehme keha animeerimisel käsitletakse kontrollpunkte nagu tippe, mistõttu suurem kontrollpunktide arv pikendab arvutusaega.
- interaktiivsel muutmisel, kuna saad liigutada ainult kontrollpunkte (kuid sa saad kontrollpunktide arvu osakeste režiimis (Particle mode) uuesti arvutada).

10 segmenti peaks olema piisav isegi väga pikkade karvade jaoks, 5 segmenti on küllalt lühemate karvade jaoks, ja 2 või 3 segmentist piisab lühikese karvastiku jaoks.

Alamosakesed

Vaata peatükki [alamosakesed](#).

Renderdamine

Karvad saab renderdada rajana, objektina või grupina. Nende kirjeldust vaata [osakeste visualiseerimise](#) peatükist.

Kasutamine



Pilt 4b: Osakeste süsteemid võivad karva kasvada...

- [Fur Tutorial](#), mille järgi on loodud *Pilt 4b*. See tegeleb spetsiaalselt lühikeste karvadega.

}

Karvade dünaamika

Karvade osakesed saab nüüd riide füüsika abil dünaamiliseks muuta.

Et karvade füüsika sisse lülitada, klõpsa nuppu välja Hair Dynamics (karvade dünaamika) kõrval.

Seaded

Material

Stiffness (jäikus)

Määrab, kui jäigad on karvade juureosad.

Mass (mass)

Määrab riidematerjali massi.

Bending (paindumine)

Määrab, kui kooldunud karv on. Kõrgemad väärtused põhjustavad vähem kooldumist.

Internal Friction (sisemine hõõrdumine)

Karvade omavaheline hõõrdumine.

Collider Friction (pörke hõõrdumine)

Hõõrdumine karvade ja väliste pörkeobjektide vahel.

Sumbumine (Damping)

Spring (vedru)

Riide liikumiskiiruse sumbumine. Kõrgem = sujuvam, vähem lainetust.

Air (õhk)

Õhul on harilikult mõningane tihedus, mis aeglustab esemete allalangemist.

Kvaliteet

Steps (samme)

Simulatsiooni kvaliteet sammudes kaadri kohta. (Kõrgem on parema kvaliteediga, kuid aeglasem).

Alamobjektid

Alamobjektid (Children) on karvadele (Hair) ja võtmeosakestele (Keyed) määratud alamosakesed. Nende abil on peamiselt võimalik töötada suhteliselt väikese hulga osakestega, mille põhjal arvutatakse füüsika. Sel juhul joonduvad alamobjektid ülemobjektide järgi. Alamobjektide arvu ja väljanägemist saab muuta füüsikat ümber arvutamata.

- Alamobjekte saab emiteerida osakestest või külgedest (mõningate erinevate võimalustega). Emissioonil külgedest (Faces) on mõned eelised: eelkõige on jaotus külgedel ühtlasem (see teeb selle sobivamaks karvade ja muu taolise jaoks). Osakeste alamobjektid seevastu järgivad paremini oma ülemobjekte: näiteks kui kasutad pehme keha animeerimist ja ei taha, et karvad läbiksid emiteerivat võret. Vaata ka meie käsiraamatu lehekülge [Karvad](#).
- Kui sa lülitad alamobjektid sisse, siis ei renderdata enam ülemobjekte (selle mõte on see, et alamobjektide kuju võib suuresti erineda ülemobjektide omast). Kui tahad lisaks näha ülemobjekte, siis lülita sisse nupp Parents (ülemad) paneelil Visualization (visualisatsioon).
- Alamobjektidel on sama material kui nende ülemobjektidel ja nende värvus oleneb täpsest asukohast, kust nad emiteeritakse (nii et kõik alamobjektid võivad olla erinevat värvi või ka teiste erinevate atribuutidega).

Võimalikud valikud sõltuvad osakeste süsteemi tüübist ja sellest, kas töötad "alamobjektidega külgedest" (*Children from faces*) või "alamobjektidega osakestest" (*Children from particles*). Me ei näita kõiki võimalikke kombinatsioone, vaid ainult seadeid karvade (Hair) osakeste süsteemi jaoks.

Seaded

Simple (lihtne)

Alamobjektid emiteeritakse ülemobjektiks olevatest karvadest.

Interpolated (interpoleeritud)

Alamobjektid emiteeritakse "ülemosakeste" vahelt võre külgedelt. Nende asukoht interpolateeritakse külgnevate ülemobjektide järgi. See on eriti kasulik karvkatte puhul, sest võimaldab ühtlast jaotust. Mõned alamobjektid võivad muutuda virtuaalseteks ülemobjektideks, mis mõjutavad lähedalolevaid osakesi.

Display (esitus)

Alamobjektide arv 3D-aknas.

Render (renderdus)

Renderdatavate alamobjektide arv (kuni 10 000).

Lihtne režiim

Size (suurus)

Ainult emiteerija (Emitter) jaoks. Alamobjektide suuruse kordaja.

Random (juhuslik)

Alamobjektide suuruse juhuslikkus.

Interpoleeritud režiim

Seed (seeme)

Nihutab juhuslike arvude tabeleid alamobjektidel, et saada erinevaid tulemusi.

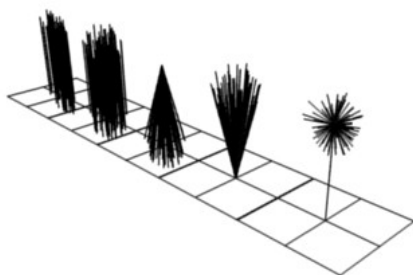
Virtual (virtuaalne)

Virtuaalsete ülemobjektide suhteline hulk.

Long Hair (pikad karvad)

Arvutab alamobjektid, mis sobivad pikkade karvade jaoks.

Efektid



Pilt 2: Vasakult paremale: Round: 0.0 / Round: 1.0 / Clump: 1.0 / Clump: -1.0 / Shape: -0.99.

Clump (puhmas)

Puhmastamine. Alamobjektide tipud võivad kokku saada ülemises otsas (1.0) või alata koos juurelt (-1.0).

Shape (kuju)

Puhma (Clump) erivorm. Kas pöördparaboolne (0.99) või eksponentsiaalne (-0.99).

Length (pikkus)

Alamobjektide pikkus

Threshold (lävi)

Alamobjektide pikkusest puutumata jäävate osakeste hulk

Radius (raadius)

Raadius, mille ulatuses Alamobjektid on ümber ülemobjektide jaotatud. Tegemist on kolmedimensiooniliste objektidega, seega alamobjekte saab emiteerida ülemobjektidest kõrgemale või madalamale.

Roundness (ümarus)

Alamobjektide ümarus ümber oma ülemobjektide. Kas kerajalt (1.0) või tasapinnaliselt (0.0).

Seed (seeme)

Nihutab juhuslike arvude tabeleid alamobjektidel, et saada erinevaid pseudojuhuslikke tulemusi.

Karedus (*Roughness*)

Uniform (ühtlane), Size (suurus)

See tugineb alamobjektide asukohale ja seega varieerib trajektoore üksteise lähedal asuvatel alamobjektidel samaselt.

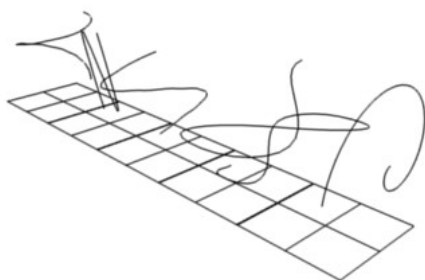
Endpoint (lõpp-punkt), Shape (kuju)

"Rough End" (kare lõpp) muudab trajektooride lõpud juhuslikuks (natuke nagu negatiivne juhuslik puhmastamine). Vorm võib varieeruda <1 (paraboolne) kuni 10.0 (hüperboolne).

Random (juhuslik), Size (suurus), Threshold (lävi)

See baseerub juhuslikul vektoril ja seega ei ole lähedalasuvate alamobjektide puhul samane. Läve saab määrata rakendumise ainult osadele alamobjektidest. See on kasulik mõne alamobjekti loomisel, mis ei käitu samamoodi nagu ülejäänud.

Vääne (*Kink*)



Pilt 3: Väändega (Kink) alamosakesed. Vasakult paremale: Curl / Radial / Wave / Braid / Roll.

Seadega Kink (vääne) saad alamobjekte keerata ümber ülemobjekti. Vaata ülalolevalt pildilt (*Pilt 3*) erinevaid väände tüüpe.

Curl (kihar)

Alamobjektid kasvavad spiraalselt ümber ülemobjektiks olevate karvade.

Radial (radiaalne)

Alamobjektid moodustavad ümber ülemobjekti laine, mis läbib ülemobjektiks oleva karvastiku.

Wave (laine)

Alamobjektid moodustavad ühesuunalise laine.

Braid (pats)

Alamobjektid palmitsevad ennast ümber ülemobjektiks oleva karvastiku.

Amplitude (ulatus)

Nihke ulatus.

Clump (puhmas)

Kui palju puhmas mõjutab väände ulatust.

Flatness (lamedus)

Karvade lamedus.

Frequency (sagedus)

Nihke sagedus (1 : kogupikkus). Mida kõrgem on sagedus, seda rohkem pöördeid tehakse.

Shape (kuju)

Väände alguskoht (väände nihe karval).

Tipugrupid (*Vertex Groups*)

Tipugruppide (*Vertex Groups*) paneel lubab sul määrata tipugrupid, mida kasutada paljude alamosakeste seadete puhul. Samuti saad iga tipugrupi mõju muuta märkeruudu abil vastupidiseks. Saad mõjutada järgnevaid atribuute:

- Tihedus (*Density*)
- Pikkus (*Length*)
- Salgulisus (*Clump*)
- Vääne (*Kink*)
- Ebaühtlus 1 (*Roughness 1*)
- Ebaühtlus 2 (*Roughness 2*)
- Lõpu ebaühtlus (*Roughness End*)

Näited

...

Osakeste režiim

Siin arutame ühte kõige imelisematest detailidest uue osakeste süsteemi juures: osakesterežiimi (Particle Mode). Selles režiimis saad sa muuta karvade (Hair) osakeste süsteemi võtmepunkte (=kontrollpunkte), nagu ka eeltöödeldud osakeste simulatsioone. Võtmepunktide arv on segmentide arv pluss üks. Lisaks saad lisada karvaosakesi (Hair *particles*).

Kuna osakeste režiimis töötamine on üsna lihtne ja väga samane tippudega töötamisele 3D-aknas, siis näitame kõigepealt, kuidas valmistada ette osakeste süsteem ja anname siis ülevaate erinevatest funktsioonidest.

Osakeste režiimi kasutusviisid

Karvaosakeste jaoks

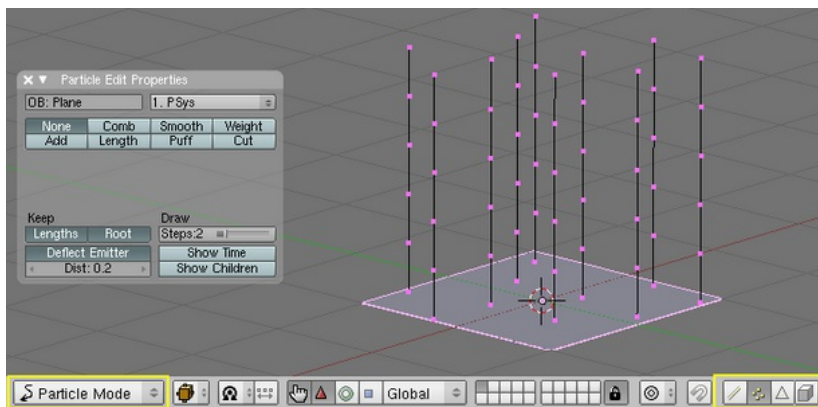
- Loo karvaosakeste süsteem.
- Anna sellele algkiirus pinnanormaali (Normal) suunas.
- Klõpsa nuppu Set Editable (määra muudetavaks).
- Vali 3D-akna päise rippmenüüst Mode osakeste režiim (Particle Mode).

Osakeste simulatsiooni jaoks

- Kasuta emiteerija (Emitter) osakesi
- Loo simulatsioon
- Eeltöötle simulatsioon
- Muuda osakeste trajektoore

Et näha, mida sa teed:

- Vali Point select mode (punktide selekteerise režiim) (3D-akna päises).
- Lülita siise paneel Particle Edit Properties (osakesterežiimi omadused, *PEP*) kiirklahviga N.




Pilt 1: Osakeste režiimi ülesseadmine.

Nüüd peaks see süsteem nägema välja selline nagu *Pildil 1*, tõenäoliselt rohkemate osakestega. Jätka ja muuda võtmepunkte.

Osakeste režiimi kasutamine

Punktide valimine

- Üksik: RMB .
- Kõik: kiirklahv A.
- Ühendatud: liiguta hiir võtmepunkti kohale ja vajuta kiirklahvi L.
- Piirdkastiga valik: B.
- Esimene/viimane: W → Select First/Select Last (vali esimene/vali viimane).

Võid kasutada ka menüüd Select (valik).

Võtmepunktide või osakeste liigutamine

- Et liigutada valitud võtmepunkte, vajuta G või kasuta mõnda paljudest tippude lohistamise meetoditest.
- Et liigutada osakest koos selle juurega, pead välja lülitama valiku Keep Root (hoia juur paigal) osakesterežiimi omaduste (Particle Edit Properties, *PEP*) paneelil.
- Saad teha paljusid neidsamu asju nagu tippudegagi, sealhulgas skaleerida, pöörata ja kustutada (terveid osakesi või võtmepunkte).
- Sa ei saa võtmepunkte või osakesi kloonida ega välja suruda, aga sa saad osakesi tükeldada, mis lisab uusi võtmepunkte (W → Subdivide (tükelda)/2 NumPad).
- Teise võimalusena saad osakese taasvõtmestada (W → Rekey (taasvõtmesta)/1 NumPad) ja valida võtmepunktide arvu.

See, kui sujuvalt karv järgib võtmepunkte, sõltub sammude (Draw Steps) arvust, mille saad seada *PEP* paneelil või visualiseerimise (Visualisation) paneelil.

Osakeste peegeldamine

- Kui tahad luua X-telje suhtes sümmetrilist soengut, pead järgima järgmisi samme:
 - Vali kõik osakesed, kiirklahv A.
 - Peegelda osakesed CtrlM või vali menüüst Particle → Mirror (osake->peegelda).
 - Lülita osakese (Particle) menüüs sisse X-Axis Mirror Editing (muutmine X-telje peegeldusega).

Võib juhtuda, et pärast peegeldamist asuvad kaks osakest peaaegu samas kohas. Kuna see oleks mälu ja renderdusaja raiskamine, võid sa eemaldada duplikaadid - Remove doubles kas täiendavate valikute menüüs (Specials) (W) või osakese (Particle) menüüst.

Peitmine/näitamine

Peitmine ja näitamine toimub osakeste puhul samamoodi nagu 3D-aknas tippude puhul. Vali üks või mitu osakese võtmepunkti, mida sa tahad peita, ja vajuta H. Tegelikult ei peitu mitte osake, vaid ainult võtmepunktid.

Peidetud osakesed (st osakesed, mille võtmepunktid on varjatud) ei reageeri erinevatele pintslitele. Aga:

Kui kasutad peegeldamist (Mirror Editing), siis liiguvad isegi peidetud võtmepunktidega osakesed, kui nende peegeldatud vasteid liigutatakse.

Valikurežiimid

Path (trajektoor)

Võtmepunktid ei ole näha, saad valida ainult tervete osakeste kaupa.

Point (punkt)

Sa näed kõiki võtmepunkte.

Tip (tipp)

Sa saad näha ja muuta (sealhulgas pintslitega) ainult osakeste tippe, st viimast võtmepunkti.

Osakesterežiimi omadused

Paneelil Particle Edit Properties (osakesterežiimi omadused), kiirklahv N, asuvad seaded karvaosakestega töötamise hõlbustamiseks.

Pintsel (Brush)

Nuppudega saad valida, mis tüüpi "kammi" tahad kasutada. Pintslitüüpide alla ilmuvad nende seaded

Ühised seaded:

Radius (raadius)

Seab pintsliradiuse.

Strength (tugevus)

Seab pintslirõhu tugevuse (ei toimi lisamisfunktsiooniga pintslil puhul).

None (puudub)

Ei ole erilist tööriista, lihtsalt muudab võtmepunkte nagu "normaalseid" tippe.

Comb (kamm)

Liigutab võtmepunkte (samane "mõjualaga muutmisele").

Smooth (silumine)

Teeb visuaalselt külgnevad segmendid paralleelseks.

Add (lisamine)

Lisab uusi osakesi.

Count (arv)

Uute osakeste arv sammu kohta.

Interpolate (interpoleeri)

Teeb uute karvade kuju olemasolevate järgi.

Steps (samme)

Pintslisammude hulk

Keys (võtmeid)

Mitme võtmepunktiga uued osakesed tehakse.

Length (pikkus)

Muudab segmentide suurust, nii et karvad on kas pikemad (Grow - kasvata) või lühemad (Shrink - kahanda).

Puff (kohevus)

Pöörab karva ümber selle esimese võtmepunkti (juure). Seega paneb see karva kas püsti seisma (Add - lisa) või pinna ligi hoidma (Sub - lahuta).

Puff Volume (kohevuse maht)

Rakendab kohevust selekteerimata lõpp-punktidele (aitab hoida karvade mahtu, kui kohevdada juuri)

Cut (lõika)

Skaleerib segmente, nii et viimane võtmepunkt ulatub pintslini.

Weight (kaal)

See on eriti kasulik pehme keha animatsiooni puhul, kuna kaal määrab pehme keha eesmärgi (Goal). Võtmepunkt kaaluga 1 ei liigu üldse, võtmepunkt kaaluga 0 allub täielikult pehme keha animatsioonile. Selle väärtuse suurust muudetakse vastavalt pehme keha eesmärgi vahemikule GMin-GMax.

Valikud

Deflect Emitter, Dist (tõuka emiteerijat, vahemaa)

Ära liiguta võtmepunkte läbi emiteeriva võre. Dist on vahemaa, mida emiteerijast hoitakse.

Keep (hoia)

Length (pikkus)

Hoiab segmentide pikkust võtmepunktide vahel, kui kammid või silud karvu. Seda tehakse kõigi teiste võtmepunktide liigutamiseks.

Root (juur)

Hoiab esimese võtmepunkti muutumatuna, et sa ei saaks karvu siirdada.

X Mirror (X peegeldus)

Lülitab sisse peegelduse üle kohaliku X-telje.

Draw (joonista)

Path Steps (trajektoori samme)

Sammude joonistamine (sama nagu visualiseerimise paneelil Visualisation).

Show Children (näita alamaid)

Joonistab lisaks osakeste alamosakesed. See lubab osakesi täpselt häälestada ja nende mõju tulemusel näha, kuid see võib muuta tegevuse väga aeglaseks, kui sul on alamosakesi palju.

Pehmed kehad



Pilt 1a: Pehme keha kujutamas riiet, mille alt ilmub välja tekst. [Animatsioon](#) – [.blend-fail](#)

Pehme keha (*Soft Body*) mõistega tähistatakse kas pehme või jäiga moonutatava keha simulatsiooni. Blenderis on see süsteem parim lihtsate riideobjektide ja suletud võrede jaoks. [Riide simulatsiooni](#) füüsika kasutab teistsugust lahendajat ja on riide jaoks parem.

Simulatsioon teostatakse, rakendades objekti tippudele või kontrollpunktidele jõudusid. On välised jõud, nagu gravitatsioon või jõuväljad, ja sisemised jõud, mis hoiavad tippe omavahel koos. Nii saad simuleerida vorme, mida objekt võtaks reaalsuses, kui tal oleks maht ja ta oleks millegagi täidetud ning talle mõjuksid reaalsed jõud.

Pehmed kehad võivad teiste objektidega vastastikku üksteist mõjutada (kokkupõrked, *Collisions*). Nad saavad mõjutada ka iseennast (kokkupõrge iseendaga, *Self Collision*).

Pehme keha simulatsiooni tulemust saab konverteerida staatiliseks objektiks. Samuti saad sa *muuta eeltöötlust*, st muuta vahepealset simulatsioonitulemust ja sealt alates simulatsiooni uuesti jooksutada.

Pehme kehade kasutamise tüüpiline stsenaarium



Pilt 1b: Tuulekott. Kott on pehme keha, nagu ka nõõrid. [Animatsioon](#) – [.blend-fail](#)

Pehmed kehad sobivad hästi järgnevate asjade jaoks:

- Elastsed objektid nii kokkupõrgetega kui ka ilma.
- Lippude ja kangaste reageerimine jõududele.
- Teatud modelleerimisülesanded, nagu padi või laudlina laual.
- Blenderil on veel teine simulatsioonisüsteem riietuse jaoks (vaata [riiete peatükki](#)). Aga sa saad mõnikord kasutada pehmeid kehasid teatud riide osade jaoks nagu laiad varrukad.
- Karvad (juhul kui sa minimeerid põrked).
- Kiikuvate kõite, kettide ja muu taolise animatsioon.

Järgnev video võib anda sulle veel mõned ideed: [\[1\]](#), [\[2\]](#)

Pehme keha loomine

Pehme keha simulatsioon töötab kõigi objektidega, millel on tipud või kontrollpunktid:

- Võred
- Kõverad
- Pinnad
- Sõrestikud

Et aktiveerida objektile pehme keha simulatsioon:

- Mine objektikonteksti (F7) ja vali füüsika (Physics) alamkontekst.
- Aktiveeri nupp Soft Body (pehme keha).

Ilmub palju valikuid. Et saada aimu kõikidest seadetest, vaata [seda lehekülge](#).

- Pehme keha simulatsiooni alustamiseks vajuta AltA.
- Pausiks vajuta Space, jätkamiseks AltA.
- Simulatsiooni peatab Esc.

Simulatsiooni kvaliteet

Seaded paneelil Soft Body Solver (pehme keha lahendaja) määravad simulatsiooni täpsuse.

Min Step (minimaalne sammude arv)

Minimaalne simulatsioonisammude arv kaadri kohta. Suurenda seda väärtust, kui pehme keha ei taba kiiresti liikuvaid pörkeobjekte.

Max Step (maksimaalne sammude arv)

Maksimaalne simulatsioonisammude arv kaadri kohta. Harilikult seatakse simulatsiooni sammude arv dünaamiliselt (Error Limit - vea piir), aga sul on arvatavasti hea põhjus selle muutmiseks.

Auto-Step (automaatsed sammud)

Määrab sammude suurused automaatselt kiiruste abil.

Error Limit (vea piir)

Määrab tulemise üldise kvaliteedi. Vaikeväärtus 0.1. Kõige kriitilisema tähtsusega seade, mis ütleb, kui täpselt lahendaja peaks pörkeid kontrollima. Alusta väärtusest, mis on 1/2 keskmisest serva pikkusest. Kui on nähtavaid vigu, võidinaid või ülemääraseid reaktsioone, vähenda selle väärtust. Lahendaja jälgib, kui "halvasti" tal läheb ja seade Error Limit (vea piir) laseb lahendajal teha mõned "adaptiivsed sammu suuruse muudatused".

Fuzzy (hägune)

Simulatsioon on kiirem, aga vähem täpne.

Choke (pidurda)

Pidurdab (vähendab väljumiskiirust) tippe või servi, mis pörke võret läbistavad.

Diagnostika

Print Performance to Console (trüki jõudlus konsoolile)

Trükitab konsoolile, kuidas lahendaja töötab.

Estimate Matrix (hinda maatriksit)

Hindab maatriksit. Jagab maatriksi asukohaks, pöördeks ja skaalaks.

Vahemälu ja eeltöötlus

Pehmed kehad ja teised füüsikalised simulatsioonid kasutavad eeltöötlusteks ja vahemäluks ühtset süsteemi. Vaata selle ülevaadet peatükist [osakeste vahemälu](#).

Simulatsiooni tulemused salvestatakse animatsiooni esitamisel automaatselt kettale, nii et järgmine kord seda käivitades loeb ta tulemusi kettalt ja mängib kiiremini. Kui sa eelarvutad (Bake) simulatsiooni, siis on vahemälu kaitstud ja sult küsitakse kinnitust, kui proovid muuta seadeid, mis teevad vajalikuks ümberarvutamise.



Ole ettevaatlik seadetega Start (algus) ja End (lõpp)

Simulatsioon arvutatakse ainult alguse (Start) ja lõpu (End) kaadrite vahel (Bake (eeltöötluste) paneelil), isegi kui sa tegelikult simulatsiooni ei eeltöötle! Seega, kui tahad pikemat simulatsiooni kui vaikimisi määratud 250 kaadrit, pead muutma kaadri End (lõpu) väärtust.

- Vahemälu kasutamine
- Kui animatsiooni esitatakse, kirjutab iga füüsikaline süsteem kõik alguskaadri ja lõpukaadri vahelised kaadrid kettale. Need failid salvestatakse .blend-failiga koos alamkaustadesse, mille nimed algavad eesliitega "blendcache".
 - Vahemälu tühjendatakse automaatselt, kui teed muudatusi - kuid mitte kõigi muudatuste puhul ja seega on mõnikord vaja seda käsitsi tühjendada. Näiteks kui oled teinud muudatusi jõuväljades. NB! Vahemälu täitmiseks peab taasesitus algama enne, või samast kaadrist, kust simulatsioon algab.
 - Kui sul ei lubata vajalikku alamkataloogi kirjutada, vahemälu salvestamist ei toimu.
 - Vahemälu saab vabastada iga füüsikalise süsteemi kohta eraldi paneelil oleva nupuga või kiirklahviga CtrlB kiirklahviga, mis vabastab kõigi valitud objektide vahemälu.
 - Sul võib tekkida probleeme, kui su .blend-faili asukoha aadress on liiga pikk ja su operatsioonisüsteemil on faili täisnime pikkus piiratud.
- Eeltöötlus
 - Pärast eeltöötlust on süsteem muudatuste eest kaitstud.
- Eeltöötluste (Bake) tulemused tühistatakse kõigi valitud objektide puhul kiirklahviga CtrlB või klõpsates nupule Free Bake (tühistab eeltöötlust) üksiku pehme keha jaoks.
 - Kui võret muudetakse, ei arvutata simulatsiooni uuesti.
- - Renderfarmide puhul on kõige parem kõik füüsikalised süsteemid eeltöödelda ja kopeerida blendcache-kaust samuti renderfarmi.

Suhtlus reaalaajas

Pehme keha simulatsiooniga töötamiseks on käepärane kasutada ajatelje (Timeline) akent. Saad vahetada kaadreid ja simulatsioon on alati näha tema tegelikus olekus. Valik Continue Physics (jätk füüsikat) ajatelje akna menüüs Playback (taasesitus) lubab sul simulatsiooniga suhelda reaalaajas, st liigutada pörkeobjekte või raputada pehme keha objekti. Ja see on päris lõbus!



Continue Physics (jätk füüsikat) ei tööta, kui mängid animatsiooni kiirklahviga AltA

See töötab ainult juhul, kui mängid animatsiooni ajatelje (Timeline) aknas nupuga Play.

Siis saad simulatsiooni jooksutamise ajal pehme keha objekti valida ja rakendada (Apply) töötleja muutmiskonteksti töötlejate

(Modifiers) paneelililil. See teeb deformatsiooni püsivaks.

Vihjed

- Pehmed kehad toimivad eriti hästi, kui tippude jaotus on ühtlane. Hea pörke jaoks on vaja piisavalt tippe. Sa muudad moonutust (jäikust), kui lisad kindlatesse piirkondadesse rohkem tippe (vaata animatsiooni pildil *Pilt 1b*).
- Pörke arvutamine võib võtta kaua aega. Kui mingi asi ei ole näha, siis milleks seda arvutada?
- Et kiirendada pörke arvutamist, on sageli kasulik lasta objektil pörgata täiendava lihtsama, nähtamatu, mõnevõrra suurema objektiga (vaata näidet pildil *Pilt 1a*).
- Kasuta pehmeid kehasid ainult siis, kui sellel on mõtet. Kui üritad katta keha võre riidetükiga ja animeerida ainuüksi pehme kehana, ei tule sellest midagi välja. Pehme keha karvade pörge iseenestega on võimalik aktiveerida, kuid seal pead sa juba üksi tee leidma. Me tegeleme [pörgetega](#) hiljem üksikasjalikumalt.
- Proovi ja kasuta sõrestiku (Lattice) või suunava kõvera (Curve Guide) pehmeid kehasid objekti enda asemel. See võib olla kordades kiirem.

Lingid

- [Arendaja märkmed](#)
- [Ahela kiikumine](#)
- [Pehmed kehad skeletiga tegelaste jaoks](#)

Välised jõud

Väliseid jõude rakendatakse pehme keha objektide tippudele (ja peaaegu ainult tippudele). See toimub Newtoni füüsikaseadusi kasutades:

1. Kui tippule ei mõju ükski jõud, siis seisab see kas paigal või liigub sirgjooneliselt muutumatu kiirusega.
2. Tipu kiirendus sõltub selle massist ja sellele mõjuvast jõust. Mida suurem on tipu mass, seda väiksem on kiirendus. Mida suurem on jõud, seda suurem on kiirendus.
3. Igale jõule on võrdne ja vastassuunaline jõud.

See toimub ainult arvutustäpsuse ulatuses, alati kasutatakse natuke summutamist, et vältida lõpmatuid arvutusi.

Näide

Me alustame väga lihtsa näitega: vaikekuubiga.

- Et otsustada välise jõu mõju üle, peaksid kõigepealt lülitama välja seade Goal (eesmärk), et tippe ei tõmmataks tagasi nende algpositsioonile.
- Vajuta kiirklahvi AltA.

Mis juhtub? Kuup liigub negatiivse Z suunas. Kõiki selle kaheksat tippu mõjutab ühtlane globaalne jõud: gravitatsioon. Gravitatsioon ilma hõõrdega on sõltumatu objekti kaalust, nii langeb siin iga objekt, mida sa kasutad pehme kehana, sama kiirendusega. Objekt ei moondu, kuna kõik tipud liiguvad sama kiirusega samas suunas.

Seaded

Pehme keha paneel

Friction (hõõrdumine)

Ümbritseva keskkonna hõõrdeegur. Mida suurem on hõõrdeegur, seda tihedam on keskkond. Kui tipp liigub ümbritseva keskkonna suhtes, tekib alati hõõrdumine.

Mass (mass)

Tippude mass. Suurem mass aeglustab kiirendust, välja arvatud gravitatsioon, kus liikumine on massist sõltumatu. Suurem mass tähendab suuremat inertsi, seega on ka pehme keha pidurdamine raskem.

Mass Vertex Group (massi tipugrupp)

Sa võid maalida tippudele võre massi väärtused ja valida siin selle tippude grupi.

Speed (kiirus)

Selle väärtusega saad muuta pehme keha sisemist ajaskaalat. See seadistab kaadrisageduse ja simulatsioonisageduse vahelise seose. Vabalt langev keha peaks läbima sekundi jooksul umbes viis meetrit. Selle seose abil saad seadistada oma stseeni mõõtkava ja simulatsiooni. Kui renderdad 25 kaadrit sekundis ja üks meeter on üks Blenderi ühik, peaksid seadma kiiruseks (Speed) 1.3.

Jõuväljad

Et luua teisi jõude, pead kasutama mõnd teist objekti - sageli kasutatakse selleks tühiobjekti Empty. Sa saad kasutada osasid jõude pehmete kehade tippudel nagu osakestel (Particles). Pehmed kehad reageerivad ainult:

- Spherical (kerajatele väljadele)
- Wind (tuulele)
- Vortex (pöörisele)

Pehmed kehad reageerivad ka harmoonilise liikumise (Harmonic) väljadele, aga mitte nii, nagu vaja. Seega, kui kasutad osakeste jaoks harmoonilist välja (Harmonic), liiguta pehme keha mõnele teisele kihile.

Vaata täpsemalt [jõuväljade](#) peatükist. Jõuväljad on küllaltki tugevad - kerajas (Spherical) väli tugevusega -1.0 tekitab umbes võrdväärse efekti gravitatsiooniga, mille tugevus on 10 njuutonit.

Aerodünaamika

Seda erilist välist jõudu ei rakendata mitte tippudele, vaid neid ühendavatele servadele. Tehniliselt rakendatakse servadele jõud, mis on risti nende suunaga. Jõud muutub koos suhtelise kiiruse projektsiooniga servale (skalaarkorutus). Pane tähele, et jõud on ühesugune nii siis, kui tuul puhub, kui ka siis, kui sa liigutad tippu sama kiirusega läbi õhu. See tähendab, et servale, mis liigub iseenda suunas, ei mõju mingit jõudu, ja servale, mis liigub iseenda suunaga risti, mõjub maksimaalne jõud.

Simple (lihtne)

Servadele mõjub hõõrdejõud ümbritsevast keskkonnast.

Lift Force (tõstejõud)

Servadele mõjub ümbritsevat keskkonda läbides tõstejõud.

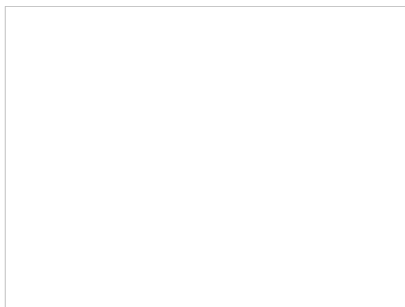
Factor (faktor)

Kui suurt aerodünaamilist jõudu kasutatakse. Proovi alguses väärtust 30.

Eesmärgi kasutamine

Eesmärk on vorm, mida pehme keha üritab võtta.

Sa pead piirama tippude liikumist mõnedes võre osades, näiteks selleks, et kinnitada pehme keha mõne teise objekti külge. Seda tehakse tipugruppide (Vertex Group) abil (sihtmärk). Sihtpositsioon on tipu algne positsioon, kus ta oleks objekti "hariliku" animeerimise korral, sealhulgas võtmevorme (Shape Keys), haake (Hooks) ja skelette (Armatures) kasutades. Tipp üritab jõuda oma sihtpositsioonile kindla, muudetava intensiivsusega.



Pilt 2b: Amortisaatori skeem.

Kujutle, et tipp on ühendatud oma sihtmärgiga vedru abil (*Pilt 2b*).

Default (vaikeväärtus)

See parameeter määrab, kui tugev selle vedru mõju on. Tugevus 1 tähendab, et tipp ei liigu üldse pehme kehana, vaid selle asemel hoiab alati oma algpositsiooni. Välja Goal (eesmärk) väärtus 0 (või üldse Goal (eesmärgi) puudumine) tähendab, et tipp liigub ainult vastavalt pehme keha simulatsioonile. Kui tippude gruppe ei ole kasutatud/määratud, näitab see numbriline väli vaikimisi eesmärgi kaalu kõikide tippude jaoks. Kui tippude grupid on olemas ja määratud, siis see nupp näitab selle asemel valikumenüüd, mis laseb sul valida eesmärgi tippugrupi nime. Kui kasutad tippugruppe, siis määrab tippu kaal eesmärgi väärtuse. Sageli kasutatakse kaalude mugavaks määramiseks [mõju maalimist](#). Objektide jaoks, mis ei ole võred, kasutatakse selle asemel nende tippude/kontrollpunktide kaalu (Weight) väärtust (kiirklahv W muutmisrežiimis (Edit mode) või teisendusomaduste (Transform Properties) paneelil. Karva (Hair) osakeste kaalu saab maalida samuti [osakeste režiimis \(Particle Mode\)](#).

Minimum/Maximum (miinimum/maksimum)

Kui maalid tippude gruppide väärtusi (kasutades mõju maalimise (WeightPaint) režiimi), võid kasutada välju G Min ja G Max kaaluväärtuste täpsemaks seadmiseks. Madalaima tippu kaalu (sinine) väärtuseks saab minimaalne (G Min) ja kõrgeimaks (punane) saab maksimaalne (G Max) (NB! Sinise-punase skaalat saab muuta kasutajaeelistuste alt).



Hetkel on kõik see rakendatud üksikutele tippudele

Me oleme käsitlenud ainult tippu liikumist sõltumata teistest tippudest, sarnaselt osakestele. Iga ilma eesmärgita (Goal) objekt kukub täielikult kokku, kui rakendada ebaühtlane jõud. Nüüd liigume edasi järgmise sammuni, milleks on jõud, mis hoiavad objekti struktuuri ja teevad pehmest kehast tõelise keha.

Stiffness (jäikus)

Vedru jäikus eesmärgi (Goal) jaoks. Madalad väärtused loovad väga nõrga vedru (elastsem "ühendus" eesmärgiga), kõrged väärtused loovad tugeva vedru (jäigem "ühendus" eesmärgiga).

Damping (summutamine)

Vedru hõõrdumine. Kõrge väärtuse korral lõpeb liikumine kiiresti (vähe vetrumist).

Sisemised jõud

Pilt 1a: Tipud ja jõud piki neid ühendavaid külgi

Et luua ühendus pehme keha objekti tippude vahel, peab olema jõud, mis hoiab tippe koos. Need jõud toimivad piki võre servi ehk tippude vahelisi ühendusi. Need jõud käituvad nagu vedrud. (*Pilt 1a*) näitab, kuidas 3×3 tippude ruudustiku tipud (tasapinna võre Blenderis) on ühendatud pehme keha simulatsioonis.

Kuid kaks tippu saavad vabalt pöörelda, kui sa ei loo nende vahele täiendavaid servi. Oled sa kunagi proovinud ehitada riulit ainult 4 lauast? Ära tee seda, see ei ole stabiilne. Loogiline meetod hoida keha kokku varisemast oleks tippude vahele täiendavate servade lisamine. See töötab päris hästi, kuid muudab drastiliselt su võre topoloogiat.

Pilt 1b: Täiendavad jõud, kui jäigad nelinurgad (Stiff Quads) on sisse lülitatud.

Õnneks laseb Blender meil määrata lisaks *virtuaalsed* ühendused. Ühest küljest saame me määrata virtuaalsed ühendused nelinurkse külje diagonaalsete tippude vahel (Stiff Quads - jäigad servad, *Pilt 1b*), teisest küljest saame me määrata virtuaalsed ühendused tipu ja iga tipu vahel, mis on ühendatud tema naabritega (Bending Stiffness - painde jäikus). Teiste sõnadega paindumismäär, mida lubatakse tipu ja iga teise tipu vahel, mis on temast kahe servaga eraldatud

Servade seaded

Servade omadused määratakse Soft Body Edge (pehme keha serv) seadetega.

Use Edges (kasuta servi)

Lubab võre objekti servadel käituda vedrudena.

Pull (tõmme)

Vedru jäikus servade jaoks (kui palju saab servi venitada). Madal väärtus tähendab väga nõrka vedru (väga elastne materjal), kõrge väärtus tähendab tugevat vedru (jäigem materjal), mis paneb laiali venitamisele vastu. 0.5 on lateks, 0.9 on nagu sviiter, 0.999 on nagu kõvasti tärgeldatud taskurätik või nahk. Pehme keha animatsioon kipub muutuma ebastabiilseks, kui kasutad väärtust 0.999, seega peaksid seda vähendama.

Push (lükkamine)

Kui palju pehme keha kannatab kokku pressimist, nagu survevedru. Madalad väärtused on kanga jaoks, kõrged aga täispuhutavate esemete ja jäikade materjalide jaoks.

Damp (summutus)

Serva vedrude hõõrdumine. Kõrged väärtused (maksimum 50) summutavad lükkamise (Push) / tõmbamise (Pull) mõju ja muudavad riide käitumise rahulikumaks.

Plastic (plast)

Pärast pörget on objekti moonutus jääv. Tipud võtavad uue asukoha ilma töötlejat rakendamata.

Bending (paindumine)

See valik loob virtuaalsed ühendused tipu ja selle naabritega ühendatud tippude vahel. See hõlmab ka diagonaalseid servi. Summutus rakendub ka neile ühendustele.

Length (pikkus)

Ühendused võivad kokku tõmbuda või paisuda. See väärtus on antud protsendina, 0 lülitab selle funktsiooni välja. 100% tähendab, et muutust ei ole, keha hoiab 100% oma suurus.

Stiff Quads (jäigad nelinurgad)

Nelinurksetel külgedel kasutatakse vedrudena ka diagonaalseid servi. See hoiab nelinurkseid külgi pörgetel täielikult kokku kukkumast (mida nad muidu teeksid).

Shear (lüke)

Nelinurksete külgede jaoks loodud vedrude jäikus.

Kokkukukkumise vältimine

Et näidata erinevate servaseadete mõju, kasutame kahte kuupi (sinine: ainult nelinurgad, punane: ainult kolmnurgad) ja laseme neil langeda tasapinnale. (Kuidas korraldada pörget, on kirjeldatud peatükis [Kokkupõrked](#)).



Pilt 3a: Kaader 1 ilma jäikade nelinurkadeta

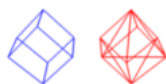


Pilt 3b: Kaader 36



Pilt 3c: Kaader 401

Pildil 3 kasutatakse vaikeseadeid (ilma jäikade nelinurkadeta). “Ainult nelinurkadest” kuup kukub täielikult kokku; kuup, mis on koostatud kolmnurkadest, hoiab oma kuju, kuigi moondub ajutiselt põrkel tekkinud jõudude tõttu.



Pilt 4a: Kaader 1, kui jäigad nelinurgad (Stiff Quads) on sisse lülitatud



Pilt 4b: Kaader 36



Pilt 4c: Kaader 401

Näites *Pilt 4* on seade Stiff Quads (jäigad nelinurgad) aktiveeritud (mõlemal kuubil). Mõlemad kuubid hoiavad oma vormi, punase kuubi puhul ei ole vahet, sest tal ei ole niikuinii nelinurkasid.



Pilt 5a: Kaader 1, kui painde jäikus (Bending Stiffness) on sisse lülitatud.

[Blend-fail](#)



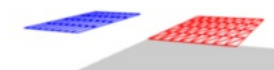
Pilt 5b: Kaader 36



Pilt 5c: Kaader 401

Teine meetod, mis aitab hoida objekti kokku kukkumast, on muuta selle painde jäikust (Bending Stiffness). See hõlmab ka diagonaalseid servi (summutamine rakendub ka neile ühendustele).

Pildil 5 on painde jäikus aktiveeritud tugevusega 1. Nüüd on mõlemad kuubid jäigemad.



Pilt 6a: Kaks tasandit kokku põrkamas



Pilt 6b: Painde jäikus puudub, kaader 101



Pilt 6c: Suur painde jäikus (10), kaader 101

Painde jäikust võib kasutada ka siis, kui tahad tükeldatud tasapinda muuta plaadisarnaseks. Ilma painde jäikuseta võivad küljed vabalt üksteise vastu keerata nagu hinged (*Pilt 6b*). Kui aktiveerid jäigad nelinurgad (Stiff Quads), siis ei muutu simulatsioonis midagi, sest selles näites ei moundu küljed üldse.

Painde jäikus hoiab ära tasapinna paindumise.

Kokkupõrked

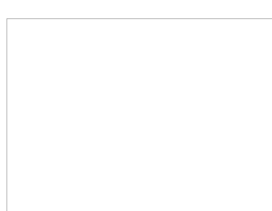
On kaks erinevat kokkupõrke tüüpi, mida sa saad kasutada: põrge erinevate objektide vahel ja sisemine põrge. Me peame ühe asja kohe alguses selgeks tegema: põrkearvutuste peamiseks sihtmärgiks on pehme keha tipud. Seega, kui sul on liiga vähe tippe, toimub liiga vähe põrkeid. Teiseks saad sa kasutada servi ja külgi põrke arvutamise parandamiseks.

Põrked teiste objektidega

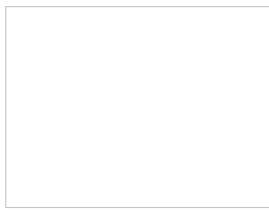
Pehme keha põrkamiseks teiste kehadega on mõned nõuded:

1. Mõlemad objektid peavad jagama kihti - kuid see ei pea olema nähtav kiht.
2. Põrkeobjekt peab olema võreobjekt.
3. Sa pead põrkeobjektile aktiveerima valiku Collision (põrge) füüsika (Physics) alamkonteksti põrke (Collision) paneelil (*Pilt 1*). Põrkeobjekt võib olla ka pehme keha.
4. Kui sa kasutad töötlejaid, nagu rida (Array) ja peegeldus (Mirror), pead sa aktiveerima valiku EV.M.Stack tagamaks, et põrke arvutamine toimub töödeldud objekti järgi. Töötlejate järjestus ei ole oluline.

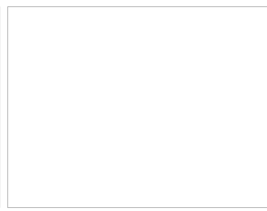
Näited



Pilt 2a: Pehme kuup põrkumas kokku tasandiga



Pilt 2b: Pehme tasand põrkumas kuubiga - vastasmõju puudub

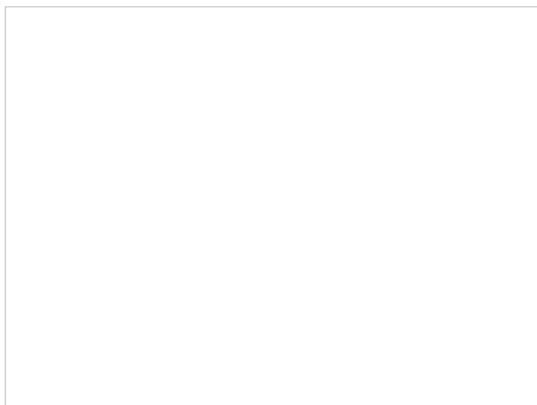


Pilt 2c: Kokkupõrge juhul, kui valik CFace on sisse lülitatud.

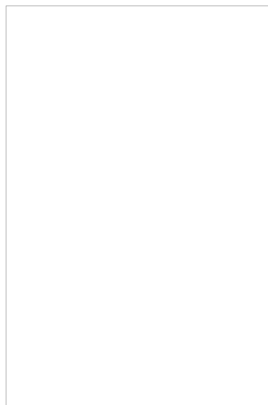
Kuup põrkumas tasapinnaga toimib päris hästi (*Pilt 2a*), kuid tasapind langeb kuubist, millega ta põrkuma peaks, otse läbi (*Pilt 2b*). Miks nii? Sellepärast, et vaikimisi kokkupõrke arvutamise meetod kontrollib ainult seda, kas mõni tasapinna neljast tipust põrkub kuubiga, kui gravitatsioon tasapinda allapoole tõmbab. Sa võid aktiveerida valiku CFace, et lülitada sisse põrge tasapinna külje ja objekti vahel (*Pilt 2c*), kuid selline arvutus võtab palju rohkem aega.

Vaatame lähemalt põrke arvutamist, et saaksid aimu, kuidas võiks seda optimeerida.

Põrgete arvutamine



Pilt 3a: Visualisatsioon pehme keha tipu põrkest tasandiga



Pilt 3b: Kuus erinevate kiirustega pehme keha tippu.
[.blend-fail](#)

Pehmete kehade simulatsioon tehakse vaikimisi tipupõhiselt. Kui pehme keha tipud ei puutu põrkeobjektiga kokku, siis ei ei mõjuta need kaks objekti teineteist üldse.

Pildil 3a näed sa tippu põrkumas tasapinnaga. Kui tipp läbib tsooni välimise (Outer) ja sisemise (Inner) vahel, tõugatakse teda pinna normaali suunas jõuga tagasi. Positsioon, kus tipp lõpuks peatub, sõltub jõududest, mis sellele mõjuvad. Näites tasakaalustavad gravitatsioon ja tagasitõuke jõud teineteist. Kiirus, millega tipp põrke tsoonist väljub, oleneb parameetri Choke väärtusest (*Pilt 4*).

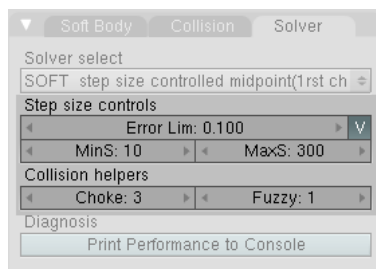
Vaatame nüüd, mis juhtub, kui teeme tipud raskemaks ja laseme neil liikuda kiiremini. *Pildil 3b* näed sa tippe liikumas erinevatel kiirustel. Kaks parempoolset (5 ja 6) liiguvad nii kiiresti, et läbib tsooni põrke tsooni (see on põhjustatud vaikimisi arvutustäpsusest - mida me saame hiljem parandada). Sa märkad, et neljas tipp liigub samuti päris kiiresti ja seetõttu murrab sisemisse tsooni. Esimesed kolm tippu põrkuvad korralikult.



Pilt 3d: Kokkupõrke arvutustes saab kasutada ka *servi* ja *külgi*

Sul on võimalik kokkupõrkel määrata, et servad ja isegi küljed võetakse põrke arvutamisel arvesse (*Pilt 3d*). Sel juhul arvutatakse põrge teistmoodi. Kontrollitakse, kas serv või külg ristub põrke objektiga, põrke tsooni sel juhul ei kasutata.

Korralik põrge



Pilt 4: Pehme keha arvutuste parameetrid

Kui põrge, mille sa oled üles seadnud, ei toimi õigesti, võid proovida järgnevat:



Parim võimalus

Lisa oma pehme keha objektile ringlõikeid (Loop Cuts) strateegilistesse kohtadesse, mis suurima tõenäosusega võtavad põrkest osa.

- Pehme keha objektile peab olema rohkem tükeldusastmeid kui põrke objektile.
- Kontrolli külje normaalide suunda.
- Kui põrke objektile on teravaid ogasid, võivad nad pehmest kehast läbi tungida.
- Põrke lahendaja arvutussamm peab vastama kiirusele, millega liiguvad pehme keha tipud. Muuda madalamaks seadet Error Lim ja suurenda ettevaatlikult Min S-i.
- Seadete Outer (välimine) ja Inner (sisemine) väärtused peksid olema küllalt suured, kuid vastaskülgede tsoonid ei tohiks kattuda, muidu tekivad sul vastassuunalised jõud.
- Kui kasutad tugevaid jõude, peaksid sul olema ka laiad tsoonid.
- Määra Choke (lämbumise) väärtus küllalt suureks (kui vaja, siis kas või lõppu välja), kui sul tekib probleeme tõrjutud tippudega.
- Põrkuvaid külgi on keeruline juhtida ja need vajavad pikka arvutusaega. Ürita neid mitte kasutada.

Sageli on parem põrkeobjektina kasutamiseks luua lihtsustatud võre, kuigi see võib olla keeruline, kui kasutad animeeritud võret.

Kokkupõrge iseenesega

Kokkupõrge iseenesega (Self Collision) töötab ainult siis, kui sa oled aktiveerinud servade kasutamise (Use Edges).

Kui see on sisse lülitatud, saad määrata, kuidas Blender väldib pehme keha iseenesega lõikumist. Iga tipp on ümbritsetud elastse virtuaalse keraga. Tipud ei saa läbistada teiste tippude kerast. Kui tahad head tulemust, on sul võib-olla vaja nende kerade suurust muuta. Tavaliselt töötab see päris hästi vaikimisi valikutega.

Ball Size Calculation (kera suuruse arvutamine)

Man (*Manual* - käsitsi)

Ball Size (kera suurus) määrab otse kera suuruse Blenderi ühikutes.

Av (*Average* - keskmine)

Arvutatakse kõikide tippuga seotud servade keskmine pikkus ja korrutatakse kera suuruse (Ball Size) seadega. Töötab hästi ühtlaselt jaotatud tippudega.

Min/Max (miinimum/maksimum)

Kera suurus vastab vähimale/suurimale vedru pikkusele selles tipus korrutatuna kera suurusega (Ball Size).

AvMiMax (*Average Min/Max* - keskmine miinimum/maksimum)

Suurus = $((\text{Min} + \text{Max}) / 2) \times \text{Ball Size}$.

Ball Size (kera suurus)

Vaikimisi 0.49 kas Blenderi ühikutes või suhtena ühendatud servade pikkusesse. Serva pikkus arvutatakse sinu poolt valitud algoritmi põhjal. Tead seda ebamugavat tunnet, kui keegi seisab sulle liiga lähedal? Me kutsume seda "isiklikuks ruumiks" ja see seade määrab faktori, mis korrutatakse vedru pikkusega. See on sfääriline distant (raadius), mille piires sinna sattunud mõni teine sama võre tipp tõugatakse eemale, et vältida põrget iseenesega.

Sea selle väärtuseks suhteline vahemaa tippude vahel, millele sa tahad anda isikliku ruumi. Liiga kõrge väärtus haarab kaasa liiga palju tippe korraga ja aeglustab arvutamist. Liiga madal tase laseb teised tipud liiga ligidale ja seetõttu võib tekkida läbitungimisi, kuna pole küllalt aega neid aeglustada.

Stiffness (jäikus)

Vaikeväärtus 1.0. Kui elastne see isikliku ruumi kera on.

Damping (summutamine)

Vaikeväärtus 0.5. Kuidas tipp reageerib. Madal väärtus ainult aeglustab tippu, kui see satub liiga ligidale. Kõrge väärtus tõukab seda tagasi.

Põrked teiste objektidega määratakse (teisel) [kokkupõrgete paneelil](#). Et põrkuda teise objektiga, peavad nad jagama vähemalt ühte ühist kihti.

Lihtsad näited

Mõned lihtsad näited, mis demonstreerivad pehmete kehade füüsika võimsust.

Põrkav kuup

Määra oma alguskaadriks *Start* 1 ja lõpukaadriks *End* 150.

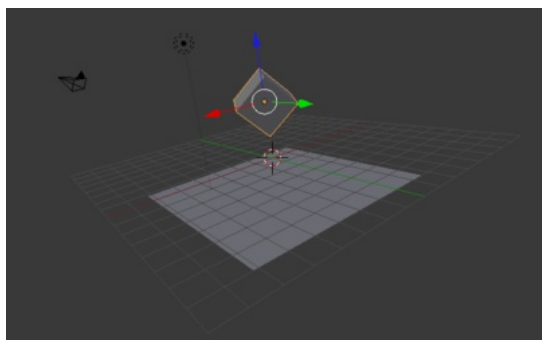


Ajatelg

Lisa tasapind ja suurenda seda viis korda. Seejärel mine füüsika sakk *Physics* ja lisa põrkumine *Collision*. Selle näite puhul vaikesätetest piisab.

Nüüd lisa kuup või kasuta stseenis juba olemasolevat kuupi. Sisene klahviga \leftrightarrow Tab muutmisrežiimi ja tükelda kuupi kolm korda. Seejärel lisa servade sujuvumaks muutmiseks sellele kantimistöötleja (*Bevel modifier*). Et selle mõju natuke suurendada, vajuta kaks korda klahvi R ja liiguta natuke oma kursorit.

Pärast seda peaks su stseen nägema välja selline:



Pehmete kehade füüsika jaoks valmis olev stseen

Kõik on pehmete kehade füüsika lisamiseks valmis. Mine füüsika sakk *Physics* ja lisa pehmekeha (*Softbody*). Võta linnuke ära eesmärgi kastist (*Goal*) ning lülita sisse pehmekeha iseendaga põrkumine (*Self collision*). Pehmekeha servade (*Soft Body Edge*) all suurenda paindumise sätet *Bending* väärtus 10-le.

Kui sa nüüd kombinatsiooniga AltA animatsiooni mängid, näidatakse sulle aeglaselt põrkavat kuupi. Et asju kiiremaks muuta, peame me pehmekeha füüsika eelrenderdama (*bake*).

Pehmekeha puhvri (*Soft Body Cache*) all muuda alguse (*Start*) ja lõpu (*End*) sätted vastavaks oma algus- ja lõppkaadritega. Antud juhul 1 ja 150. Testimaks, kas kõik töötab, võid määrata puhvisammuks (*Cache Step*) 5 või 10, kuid lõpliku animatsiooni puhul on parem vähendada see 1-ks, et puhverdataks kõik.

Pärast seda peaks su füüsika paneel *Physics* nähema välja selline:



Füüsika sätted.

Nüüd saad simulatsiooni eelrenderdada, lisada kuubile materjalid ja tekstuudid ning seejärel terve animatsiooni renderdada.

Tulemus

Renderdatud põrkav kuup:

[\[video link\]](#)

Riide simulatsioon



Riide näidis



Riie nikerdatud puufiguuridel (autor motorsep).



Riide näidis

Riide simulatsioon on üks raskemaid arvutigraafika aspekte, kuna riie on petlikult lihtne reaalse maailma objekt, mida võetakse enestmõistetavana, kuid millel on tegelikult väga keerulised vastasmõjud iseenda ja keskkonnaga. Pärast aastepikkust arendamist on Blenderil väga töökindel riide simulaator, mida kasutatakse rõivaste, lippude, vimplite jms jaoks. Riidet mõjutavad liikuvad objektid, tuul ja teised jõud, samuti stseeni üldine aerodünaamiline mudel, mis kõik on sinu kontrolli all.

Kirjeldus

Riidetükk võib olla igasugune võre, avatud või suletud, mis on riidena määratud. Riide (Cloth) paneelid asuvad füüsika (Physics) alamkontekstis ja koosnevad kolmest seadete paneelist. Riie on kas avatud või suletud kaalutu võre – st eeldatakse, et ühel riideobjektile on kogu ulatuses sama tihedus (sama mass pinnahüki kohta).

Riie modelleeritakse harilikult ruudustikvõre või kuubina, kuid ta võib olla ka näiteks kas või kaisukaru. Siiski, Blenderi [pehmete kehade süsteem](#) (*Soft Body*) pakub suletud võrede jaoks paremat simulatsiooni. Riie on spetsialiseeritud kangaste jaoks.

Kui objekt on määratud riidena, lisatakse tema töötlejate pinusse automaatselt [riidetöötleja](#) (*Cloth Modifier*). [Töötlejana](#) toimib ta koosmõjus teiste töötlejate, näiteks skeleti (Armature) ja silujaga (Smooth). Nendel juhtudel arvutatakse võre lõplik vorm vastavalt töötlejate järjestusele pinus. Näiteks peaks riide siluma alles *pärast* seda, kui töötleja on arvanud tema kuju.

Riidet saad muuta kahes kohas. F7 füüsika nuppudega, mille abil saab muuta riide omadusi, ja töötlejate pinus, kus saad muuta riidetöötleja omadusi, mis on seotud väljanägemise ja koosmõjuga teiste töötlejatega.

Sa saad riidetöötlejat rakendada (Apply) võre külmutamiseks või paigalelukustamiseks selles kaadris, mis eemaldab töötleja. Näiteks võid sa laotada lina laua kohale, lasta simulatsioonil töötada ning seejärel rakendada töötleja. Sel juhul kasutad sa simulaatorit, et säästa kõvasti modelleerimisaega.

Simulatsiooni tulemused salvestatakse vahemällu, nii ei pea võre vormi, mis on animatsioonis iga kaadri jaoks ühe korra arvutatud, enam uuesti arvutama. Kui simulatsioonis tehakse muudatusi, on kasutajal täielik kontroll vahemälu puhastamise ja simulatsiooni taasjooksutamise üle. Simulatsiooni esimest korda jooksutamine on täiesti automaatne ja eeltöötlus ega erinevad sammud ei katkesta tööd.

Riide vormi arvutamine igal kaadril on automaatne ja toimub taustaprotsessina: nii saad simulatsiooni arvutamise ajal tööd jätkata. Siiski nõuab see tegevus olulist protsessorivõimsust ja sõltuvalt sinu arvuti kiirusest ja simulatsiooni keerukusest võib protsessori koormamine muuta Blenderi märgatavalt uimasemaks.

Ära hüpka edasi

Kui sead riide simulatsiooni valmis, kuid Blender ei ole veel arvanud vorme kogu simulatsiooni kaadri vahemiku jaoks ja sa hüppad oma animatsioonis palju kaadreid edasi, võib juhtuda, et riide simulaator ei suuda arvutada või näidata täpset võre vormi sellel kaadril, kui ta ei ole enne arvanud vormi eelmis(t)es kaadri(t)es.

Töö järjekord

Üldine protsess riidega töötamisel on:

1. Modelleeri riideobjektile üldine algusvorm.
2. Määra objekt objektikonteksti Object füüsika alamkontekstis Physics "riidena" (F7).
3. Modelleeri teised suunavad objektid, mis riidet mõjutavad. Tee kindlaks, et suunav töötleja (*Deflection*) on töötlejate järjestuses viimane, pärast teisi võret moonutavaid töötlejaid.
4. Valgusta riie ja määra materialid ja tekstuurid, tee UV-lahtilõige, kui soovid.
5. Kui soovid, anna objektile osakesed, nagu näiteks pinnalt kerkiv aur.
6. Jooksuta simulatsioon ja sedista valikud, et saada rahuldav tulemus. Ajagraafiku akna nn makinupud on selle sammu jaoks suurepärased.
7. Soovi korral peata võre mingis simulatsiooni punktis, et anda talle uus vaike-algusvorm.
8. Tee võres kaaderhaaval väiksemaid muudatusi, et parandada väikseid rebendeid.

Riidesimulatsioonide loomine

Selles lõigus selgitatakse, kuidas valida seaded soovitud tulemuse saamiseks. Esiteks, lülita sisse Cloth (riie). Määra, mis sorti riidet simuleerid. Võid valida alustamiseks ühe eeldefineeritud riidesortidest.

Nagu näed, mida raskem on kangas, seda jäigem see on, seda vähem venib ja seda vähem mõjutab seda õhk.

Riide paneel

Presets (eelseadistused)

Sisaldab mitut eelvalmistatud riidenäidist ja laseb sul lisada enda omi.

Quality (kvaliteet)

Seadista simulatsiooni sammude arv kaadri kohta. Kõrgemad väärtused annavad tulemuseks parema kvaliteedi, kuid on aeglasemad.

Materjal

Mass (mass)

Riide materjali mass.

Structural (struktuurne)

Riide üldine jäikus.

Bending (paindumine)

Kortsude koefitsient. Kõrgem väärtus tekitab rohkem suuri kortse.

Sumbumine (Damping)

Spring (vedru)

Riide liikumiskiiruse sumbumine. Kõrgem = sujuvam, vähem lainetust.

Air (õhk)

Õhul on harilikult mõningane tihedus, mis aeglustab esemete allalangemist.

Kinnitamine (Pinning)



Riidesimulatsioon

Esimene asi, mida sa riide kinnitamiseks vajad, on [tipugrupid](#) (*Vertex Groups*). Nende tegemiseks on erinevaid võimalusi, sealhulgas mõju maalimise tööriist (Weight Paint), mis võimaldab maalida alad, mida soovid kinnitada (vaata käsiraamatust [mõju maalimise](#) osa).

Kui tipugrupid on loodud, on asjad edasi üsna lihtsad, sa ei pea tegema muud kui vajutama nuppu Pinning of cloth (riide kinnitamine) paneelil Cloth (riie) ja valida, millist tippude gruppi tahad kasutada ja kui suurt jäikust sa sellele tahad.

Stiffness (jäikus)

Sihtmärgi asukoha jäikus. Jäikuse võid jätta nii nagu ta on, vaikimisi väärtus 1 on sobiv.

Kokkupõrked

Reeglina riidetükk ei ripu niisama 3D-ruumis, vaid põrkub vastu teisi keskkonnas olevaid objekte. Et kindlustada simulatsiooni õige toimimine, on vaja mitu asja seadistada ja koos tööle panna:

1. Riide objekt peab olema määratud osalema kokkupõrgetes.
2. Võid (soovitavalt) lasta riidel põrkuda ka riide endaga.
3. Teised objektid peavad olema riide objektile jagatud kihtide *abil* nähtavad.
4. Teised objektid peavad olema võreobjektid.
5. Teised objektid võivad liikuda või olla moonutatud kolmandate objektide poolt (nagu skelett või võtmevorm).
6. Teistele võreobjektidele tuleb öelda, et nad mõjutaksid riideobjekti.
7. Tööfail (.blend) tuleb salvestada kataloogi, et simulatsiooni oleks võimalik salvestada.
8. Seejärel tuleb simulatsioon eeltöödelda (Bake). Simulaator arvutab riide kuju etteantud kaadrite vahemikus.
9. Seejärel saad muuta simulatsiooni tulemusi või teha osades kaadrites täpsustusi riidevõres.
10. Sa saad teha muudatusi keskkonnas või moonutada objekte ning seejärel taaskäivitada riidesimulatsiooni käesolevast kaadrist alates.

Kokkupõrke seaded

Riide kokkupõrke (Cloth Collision) paneel.

Nüüd pead sa ütleva riideobjektile (Cloth), et tahad selle osalemist põrgetes. Otsi riideobjekti jaoks üles paneel Cloth Collision (riide kokkupõrked), mis on näidatud paremal:

Enable Collisions (luba kokkupõrkeid)

Klõpsa sellel **LMB** , ütlema riideobjektile, et ta peab eest liikuma.

Quality (kvaliteet)

Üldine seade, et määrata, kui kvaliteetne simulatsioon on. Suurematel numbritel on see aeglasem, kuid kindlustab, et rebendeid ja riidest läbitungimisi esineb vähem.

Distance (vahemaa)

Kui teine objekt satub riidele nii lähedale (Blenderi ühikutes), hakkab simulatsioon riid eest ära lükkama.

Repel (tõuka eemale)

Tõukumisjõud, mida rakendatakse, kui riie on lähedal pörkele.

Repel Distance (tõukumisvahemaa)

Maksimaalne vahemaa tõukumisjõu rakendamiseks. Peab olema suurem kui minimaalne vahemaa.

Friction (hõõrdumine)

Koefitsient, mis määrab, kui libe on riie teiste võreobjektidega pörkudes. Näiteks siidil on madalam hõõrdekoefitsient kui puuvillal.

Kokkupõrked iseenesega

Päris riie ei tungi iseendast läbi, seega harilikult tahad sa, et riie pörkuks ka enesega.

Enable Self Collisions (lülita pörked iseenesega sisse)

Klõpsa seda, ütlema riidele, et ta ei läbistaks iseennast. See lisab küll simulatsioonile arvutusaega, kuid kindlustab realistlikuma tulemuse. Lipul, mida vaadatakse eemalt, ei ole vaja seda sisse lülitada, kuid lähivaatel keebist või pluusist tegelase seljas on see vajalik.

Quality (kvaliteet)

Et suurendada enesepörke kvaliteeti, suurenda lihtsalt välja Quality väärtust ning simulatsioon arvutab rohkem enesepörke kihte. Pea ainult mees, et väljade Collision Quality (kokkupörke kvaliteet) ja Quality (kvaliteet) väärtused peaks olema vähemalt samad.





Distance (vahemaa)

Kui tekib probleeme, võid muuta ka välja Min Distance (minimaalne vahemaa) väärtust enesepörke jaoks. Parim väärtus on 0.75; kiirete asjade jaoks pane parem 1.0. Väärtus 0.5 on küllaltki riskantne (arvatavasti on palju läbitungimisi), kuid töötab kiiremini.

Regressioonitesti fail: [Riide enesepõrked](#).

Jagatud kihid

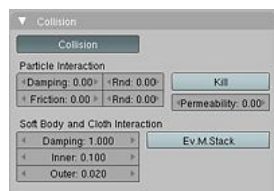
Oletame, et sul on kaks objekti: paar pükse kihitel 2 ja 3 ja tegelase võre kihitel 1 ja 2. Oled pükstel lülitanud eelpoolkirjeldatud moel sisse riidesimulatsiooni. Nüüd pead sa tegelase tegema riideobjektile "nähtavaks", nii et kui tegelane painutab jalga, tõukab see riid. See põhimõte on sama kõikidel simulatsioonidel: simulatsioon toimib ainult objektidega, mis jagavad mõnda kihti. Selles näites jagavad mõlemad objektid kihti 2.

Et vaadata/muuta objekti kihte, selekteeri objekt, klõpsates **RMB**  3D-vaate objektirežiimis. Too kiirklahviga M esile hüpikaken "kihtide liigutamine" (*Move layers*), mis näitab kõiki kihte, kus objekt asub. Et asetada objekt ühele kihile, klõpsa **LMB**  vastava kihi nupul. Et asetada objekt mitmele kihile, klõpsa **Shift LMB**  igal vastava kihi nupul. Et eemaldada objekt valitud kihilt, klõpsa lihtsalt uuesti **Shift LMB**  vastava kihi nupul, et see välja lülitada.

Võreobjektide kokkupõrked

Kui pörkuv objekt ei ole võreobjekt, nagu näiteks NURBS-pind või tekstiobjekt, siis tuleb ta kõigepealt teisendada võreobjektiks. Et seda teha, selekteeri ta objektirežiimis ja vali 3D-vaate päises Object → Convert Object Type (objekt->konverteeri objektitüüp) (AltC) ning seejärel võta hüpikaknast Mesh (võre).

Riide/objekti pörked



Kokkupõrke seaded

Riideobjekti peab mõni teine objekt tõukama. Et riid tõugata, peab sellel objektil olema sisse lülitatud kokkupõrge riidega. Et lülitada sisse riide/objekti pörked, pead sa suunamise sisse lülitama pörkuval objektil (mitte riideobjektil).

Otsi nuppude Buttons objektikonteksti Object füüsika alamkonteksti Physicspaneelil küljel üles paneel Collision (kokkupõrked). Samuti on oluline tähele panna, et selle pörkepaneeli abil öeldakse kõikidele simulatsioonidele, et see objekt on osaline pörgetes ja teiste objektide mõjutamisel jagatud kihtide kaudu (osakesed, pehmed kehad ja riie).

Pane tähele

Kokkupõrgete (Collision) paneeli on kolm erinevat, mis kõik asuvad füüsika (Physics) alamkontekstis. Vaikimisi on esimene neist kaart väljade (Fields) paneeli kõrval, ja seda meil vaja ongi. Teine paneel, kaart pehme keha (Soft Body) grupis, käib pehmete kehade kohta (riidega ei ole sellel mingit tegemist). Ja viimast oleme me juba näinud: vaikimisi on see kaart riide (Cloth) paneeli kõrval.

Võreobjekti töötlejate pinu



Kokkupõrgete pinu

Objekti kuju deformeerib riid, seega peab riide simulatsioon teadma võreobjekti "tegelikku" kuju antud kaadris. See tõeline kuju on põhivorm, mida on muudetud võtmevormide või skelettide poolt. Seetõttu peab kokkupõrke (Collision) töötleja tulema **pärast** kõiki neid teisi töötlejaid. Pilt paremal näitab tegelase võreobjekti (mitte riideobjekti) töötlejate (Modifiers) paneeli.

Riide vahemälu

Vahemälu seaded riide jaoks on samad mis teistel dünaamilistel süsteemidel. Detailidest loe peatükist [Osakeste vahemälu](#).

Pörke eeltöötlus



Pärast seda, kui oled seadistanud suunava võre kaadrite vahemikus, mille vahel kavatsed simulatsiooni jooksutada (kaasa arvatud võre animeerimine skeleti abil), võid käskida riide simulatsioonil arvutada (ja vältida) kokkupõrked. Vali objektikonteksti füüsika alamkontekstis riideobjekt, seadista alguse (Start) ja lõpu (End) kaadrid, mille vahel tahad simulatsiooni arvutada, ja klõpsa nuppu Bake (eeltöötlus).

Sa ei saa muuta alguse ega lõpu väärtust, ilma et tühjendaksid simulatsiooni eeltöötluse. Kui simulatsioon on lõpetatud, märkad sa, et sul on valikud: vabasta eeltöötlus, muuda eeltöötlust või eeltöötle uuesti.

On mõned asjad, mida sa arvatavasti kohe tähele paned. Esiteks eeltöötleb ta märgatavalt aeglasemalt kui enne ja lõikub sageli võrega.

Vahemälu oleva simulatsiooni muutmine

Vahemälu sisaldab võre kuju igas kaadris. Sa saad muuta vahemälu olevat simulatsiooni, kui vajutasid eeltöötluse käivitamise järel nuppu Bake Editing (eeltöötluse muutmine). Mine lihtsalt kaadri, mida tahad parandada, ja sisene klahviga \Leftarrow Tab muutmisrežiimi (Edit mode). Seal saad liigutada tippu, kasutades kõiki Blenderi võre vormimise tööriistu. Kui väljud, säilitatakse võre vorm sellel animatsiooni kaadril. Kui tahad, et Blender jätkaks simulatsiooni, kasutades edasi uut vormi, klõpsa LMB $\left[\right]$ nupul Rebake from next Frame (eeltöötle uuesti järgmisest kaadrist) ja esita animatsioon. Sel juhul Blender jätkab simulatsiooni, kasutades uut võre kuju.

Muuda võret, et parandada väiksemad rebendid ja kohad, kust pörkuv objekt on riid läbistanud.

Kui sa lisad, kustutad, surud välja või eemaldad võrest tippu, siis arvestab Blender uut võret kui algset võre kuju animatsiooni "esimese kaadri" välja, asendades algse vormi, millega sa alustasid, kuni kaadri, kus sa olid, kui võret muutsid. Seega, kui sa muudad võre ülesehitust, siis lahkudes \Leftarrow Tab abil muutmisrežiimist, peaksid sa tühjendama vahemälu, , et Blender teeks kokkusobiva animatsiooni.

Probleemide lahendamine

Kui sul tekib probleeme pörketuvastamisega, on nende lahendamiseks kaks meetodit.

- Kiireim meetod on muuta minimaalse vahemaa (Min Distance) seadet riide kokkupõrke (Cloth Collision) paneelil. See on küll kiireim meetod läbistamise korrigeerimiseks, kuid on vähem täpne ja ei näe nii hea välja. Selle meetodi kasutamisel kipub riidest jääma mulje, nagu see ripuks õhus, üldse on üldmulje väga ümara väljanägemisega.
- Teine meetod on suurendada kvaliteeti (Quality) (esimesel riide (Cloth) paneelil). Selle tulemuseks on väiksemad sammud simulaatori jaoks ning seetõttu on suurem võimalus, et kiiresti liikuvad pörked registreeritakse. Samuti võid suurendada pörke kvaliteeti (Collision Quality), et pörke lahendamisel tehtaks rohkem iteratsioone.
- Kui ükski nendest meetoditest ei aita, saad hiljem hõlpsasti eeltöödeldud tulemust muutmisrežiimis (Edit mode) parandada.
- Moonutav võre rebestab minu riid: suurenda struktuuraset jäikust (väli StructStiff riide (Cloth) paneelil) väga kõrge väärtuseni, näiteks 1000.

Subsurf ehk pinnatükeldaja

Eeltöötlus tehakse iga pinnatükeldamise taseme kohta, seega kasuta nii renderdamisel kui ka eelvaatel **sama** pinnatükelduse taset.

Näited

Et riidega alustada, on esimese asjana vaja muidugi kangast. Seega kustutame vaikselt loodud kuubi ja lisame tasapinna. Mina suurendasin enda oma mõõda Y-telge, kuid see ei ole vajalik. Et saada head lõtva ja elastset kangast, on vaja seda mitu korda tükeldada. Selle näite jaoks tegin ma seda 8 korda. Seega mine kiirklahviga ⇄ Tab muutmisrežiimi} vajuta W → Subdivide multi (tükelda mitmeks) ja pane selle väärtuseks 8.

Nüüd teeme sellest riide, minnes objektikonteksti (Object context) (F7) → Physics (füüsika) alamkonteksti. Keri alla, kuni näed riide (Cloth) paneeli, ja vajuta nuppu Cloth (riie). Ilmub palju seadeid, millest enamikku me praegu ignoreerime.

See on kõik, mis on vaja teha, et seada riie valmis animeerimiseks, aga kui vajutad nüüd kiirklahvi AltA, siis su suurepärane riie lihtsalt langeb väga ebasuurepäraselt. Seda me käsitleme järgmises kahes lõigus kinnitamisest ja pörkumisest.

Simulatsiooni kasutamine võre vormimiseks/voolimiseks

Sa võid igal hetkel rakendada (Apply) riide (Cloth) töötleja, et külmutada võre kuju sellel kaadril. Seejärel võid riide taas sisse lülitada, seades alguse ja lõpu kaadrid, millest alates simulatsiooni edasi jooksutada.

Teine näide ajastamisest on lipp. Määra lipp lihtsa ruudustikuna ja kinnita serv lipuvarda külge. Simuleeri umbes 50 kaadrit ja lipp langeb rippuvasse "puhkeasendisse". Rakenda riidetöötleja (Cloth modifier). Kui tahad, et lipp lehviks või muudmoodi liiguks, lülita see taas sisse nende kaadrite jaoks, mil ta on kaamera vaateväljas.

Riide silumine

Kui oled järelnud samm-sammult õpetusi eelmises osas, siis on su riie arvatavasti natuke nurgeline. Et muuta seda kenaks ja siledaks nagu pildil, on sul vaja rakendada silumistöötleja (Smooth) ja/või pinnatükeldaja (Subsurf) muutmiskonteksti töötlejate (Modifiers) paneelil (F9). Seejärel leia samas kontekstis paneel Links and Materials (seosed ja materjalid) (sama, mida sa kasutasid tipugruppide) ja vajuta nuppu Set Smooth (muuda siledaks).

Kui sa nüüd vajutad kiirklahvi AltA, hakkab asi päris ilus välja nägema, kas pole?

Riie skeletil

Riie, mida moonutab skelett ja lisaks veel pörkuv objekt: [Regressioonitesti .blend-fail](#).

Riide kasutamine pehmete kehadena



Riide kasutamine pehmete kehadena

Riie saab kasutada ka pehmete kehade simuleerimiseks. See ei ole küll riide põhieesmärk, kuid töötab sellegipoolest. Näidispiilt kasutab standardset kummimaterjali (Rubber) ilma seadeid muutmata: lihtsalt AltA.

Blenderi fail näidispiildi jaoks: [Riide kasutamine pehmete kehadena](#).

Riie ja tuul



Lipp, millele mõjub tuul

Regressioonitesti .blend-fail riide jaoks koos tuulega ja iseenesega pörgetega (samuti .blend fail ülaloleva pildi jaoks): [Riideest lipp, millele mõjub tuul ja pörked iseenesega](#).

Vedeliku simulatsioon

Mode: Objektirežiim (Object Mode) / muutmisrežiim (Edit Mode) (võred)

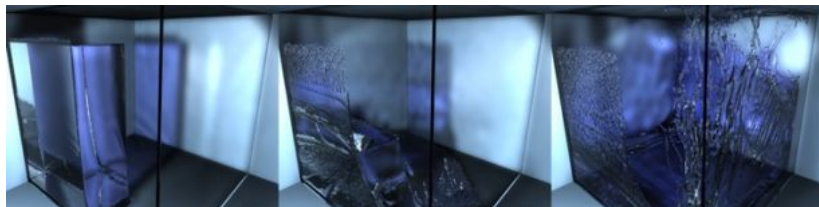
Panel: Füüsika (physics) allkontekst → Fluid (vedelik)

Hotkey: F7

Kirjeldus

Modelleerides Blenderis stseeni, saab määrata osad objektid võtma osa vedeliku simulatsioonist, näiteks kas siis vedeliku või takistusena. Teise objekti piirdkasti kasutatakse, et määrata risttahuka kujuline ala, milles simulatsioon toimub (nn "simulatsiooni toimumispiirkond"). Toimumispiirkonda määravale objektile saab seada üldised simulatsiooni parameetrid (nagu viskoossus ja gravitatsioon).

Kasutades eeltötluse (BAKE) nuppu, eksporditakse geomeetria ja seaded simulaatorisse ja teostatakse vedeliku simulatsioon, genereerides vedelikupinna võre koos eelvaatega iga animatsiooni kaadri jaoks, ja salvestatakse need kõvakettale. Siis laaditakse hetkel käsilolevale kaadrile vastav vedeliku pind kettalt ja kuvatakse või renderdatakse.



Puruneva tammi näide

Protsess

Üldiselt järgid sa järgmisi samme:

1. Modelleeri stseen (objektid, materjalid, valgus, kaamera).
2. Määra see osa stseenist, kus hakkab vedelik voolama (toimumispiirkond).
3. Täpsusta teiste vedelikuga seotud objektide funktsioonid (sissepääs, väljapääs, takistus jne).
4. Loo vedeliku allikas (allikad) ja täpsusta selle (nende) materjal, viskoossus ja algiirius.
5. Eeltötle esialgne simulatsioon.
6. Tee vajalikud muudatused ja salvesta.
7. Eeltötle lõplik simulatsioon.

Kui eeltötlus simulatsiooni, siis muutub versioonitunnus kasutajaeelistuste (User Preferences) akna päises olekuribaks, mis näitab eeltötluse edenemist. Eeltötlus nõuab **palju** arvutusvõimsust (ja aega). Seda ei ole mõtet jälgida, sest nagu öeldakse: vesi, mida jälgid, ei hakka keema, ja kook, mida vaatad, ei küpse; ja vedeliku eeltötluse vaatamine on nagu jälgida muru kasvamist. Eeltötlust on parim teha üleöö.

Stseenidevahelise ülekandmise ja linkimise võimalikkuse tõttu saab olla ühe .blend-faili kohta ainult üks toimumispiirkond.

Valikud



Blender 2.5 vedelikusimulatsiooni seaded, valitud on Domain (toimumispiirkond)

Domain (toimumispiirkond)/Fluid (vedelik)/Obstacle (takistus)/Inflow (sissevool)/Outflow (väljavool)/Particle (osake)/Control (juhtimine)
Nendest nuppudest ühe valimine määrab, mismoodi valitud objekti simulatsioonis kasutatakse. Igaüks nendest nuppudest määrab erineva funktsiooni ja vastavalt ka erinevad edasised valikud.

Võre renderdamine

Kui võrel on töötlejad, kasutatakse võre eksportimisel vedelikuarvutajasse renderdamisseadeid. Olenevalt seadetest võib arvutusaeg ja mälu kasutus plahvatuslikult kasvada. Näiteks kasutades liikuvat võret koos pinnatükeldajaga (Subsurf) takistusena, võib pinnatükelduse väljalülitamine või tasemete vähendamine aidata vähendada simulatsiooniaega. Kui seaded/liigendus on sellised nagu vaja, võid hiljem alati suurendada seadeid, et saavutada realistlikum tulemus.

Toimimispiirkond (*Domain*)

10 cm kruus lahutusega 70

10 cm kruus lahutusega 200

Objekti piirdkast määrab simulatsiooni piirid. **Kõik vedelikobjektid peavad olema toimimispiirkonna sees.** Sellest väljaspool olevaid vedelikobjekte ei eeltöödelda. Väljapoole toimimispiirkonda ei saa sattuda mitte ühtegi tilka, nähtamatud jõuväljad hoiaks nagu vedelikku 3D-ruumis. Stseenis saab olla ainult üks vedeliku simulatsiooni toimimispiirkonna objekt.

Ei ole vahet, mis kujuga on see objekt, sest seda koheldakse *alati* kui kasti (piirdkasti küljed võivad olla erineva pikkusega). Seega ei ole harilikult vaja kasutada muud keha kui kast. Kui vajad vedeliku voolu mõjutamiseks takistusi või teisi piire peale kasti, siis pead lisama täiendavaid takistusobjekte toimimispiirkonna piiride sisse.

See objekt "asendatakse" simulatsiooni käigus vedelikuga.

Lahutus (*Resolution*)

Detailsus, millega tegelik vedeliku simulatsioon läbi viiakse. See on simulatsiooni juures ilmselt kõige olulisem seade, kuna määrab vedeliku detailsuse, mälu ja ketta kasutuse ning samuti ka arvutusaja.

Pane tähele, et vajamineva mälu hulk kasvab kiiresti: lahutus 32 vajab umbes 4 MB, 64 umbes 30 MB, seevastu 128 vajab juba rohkem kui 230 MB. Olenevalt sellest, kui palju mälu sul on, veendu, et resolutsioon on piisavalt madal, et hoida Blenderit hangumast või kokku jooksmast. Pea meeles, et mitmetel operatsioonisüsteemidel on piiratud see, kui palju mälu ühele "protsessile", nagu näiteks Blenderile eraldatakse, isegi kui "masinal" on mälu sellest rohkem. Tee kindlaks, mis on sinu arvuti piirangud.



Toimimispiirkond maailmas

Veendu, et seadistad lahutuse parajaks toimimispiirkonna suuruse jaoks reaalses maailmas (Toimimispiirkonna tegelik füüsiline suurus meetrites määratakse vedeliku kastikese all asuval väljal *Domain World* (Toimimispiirkond maailmas)). Kui toimimispiirkond ei ole kuubikujuline, määratakse lahutus pikima külje järgi. Teiste külgede lahutust vähendatakse vastavalt nende pikkusele (seetõttu vajab sama resolutsiooniga mittekuubikujuline toimimispiirkond vähem mälu kui kuubikujuline).

Eelvaate lahutus (*Preview Res*)

See on lahutus, millega genereeritakse pinnavõrede eelvaade. Seega ei mõjuta see tegelikku simulatsiooni. Isegi kui eelvaates "ei ole midagi näha", võib seal olla õhuke vedelikupind, mida eelvaates ei ole võimalik näidata.

Alguse ja lõpu aeg

Start (algus) on simulatsiooni algusaeg (sekundites). See valik paneb simulatsiooni arvutamise Blenderis algama tegeliku simulatsiooni algusest hiljem. Toimimispiirkonna deformatsioone ja vedeliku voolamist, mis eelnevad algusajale, ei salvestata.

Näiteks kui tahad, et vedelik ilmuks alles siis, kui ta on enne tegelikku esimest kaadrit juba 4 sekundit voolanud, peaksid sisestama siia 4.0.

End (lõpp) on simulatsiooni lõpu aeg (sekundites).

Alguse ja lõpu aegadel ei ole mingit tegemist sellega, kui mitu kaadrit eeltöödeldakse - need põhinevad füüsikalisel jõul ja vedeliku viskoossusel.

Eeltöötlus algab alati kaadris 1:

Vedeliku simulaator eirab seadet Sta seadet animatsiooni paneelil ja eeltöötlus algab alati kaadrist 1.

Kui soovid, et simulatsioon algaks hiljem kui kaader 1, pead sa muutma vedelikobjektid oma toimispiirkonnas mitteaktiivseks

kuni kaadrini, millest alates sa tahad simulatsiooni alustada. Täpsema informatsiooni saamiseks loe [altpoolt](#).

Eeltöötlus lõpeb alati lõpukaadril End, mis on määratud animatsiooni paneelil (Anim):

Kui su kaadrisagedus on 25 kaadrit sekundis ja lõpu aeg on 4.0 sekundit, siis sa peaksid (kui alguse aeg on 0) seadma animatsiooni lõppema kaadril $4.0 \times 25 = 100$.

“Alguse aeg” ja “lõpu aeg” on väljendatud “simulatsiooniajas” (sekundites):

Kui sead oma algusajaks (Start) 3.0 ja lõpuajaks (End) 4.0, siis simuleerid sa vedeliku liikumist 1 sekundi jooksul. See üks sekund jaotatakse nii mitme kaadri vahel, nagu sa seadsid animatsiooni paneeli väljal End (lõpp). See tähendab, et kui sa näiteks oled Blenderi seadnud tegema 250 kaadrit kaadrisagedusel 25 kaadrit sekundis (Scene (stseenikontekstis) → Render alamkontekstis → paneelid Anim (animatsioon) ja Output (väljund)), paistab nagu oleks vedelik simulatsiooni alguseks juba kolm sekundit voolanud, *kuid* liigub edasi aeglustatult (üks kümnendik normaalsest kiirusest), sest kümnes sekundis videos on üks sekund simulatsiooni. Et seda parandada, muuda lõpu aeg $3.0 + 10.0 = 13.0$, et see vastaks 250 kaadrile sagedusel 25 kaadrit sekundis. Nüüd on simulatsioon reaajas, sest sa seadsid 10 sekundit vedeliku liikumise simulatsiooni 10 sekundile animatsioonile. Need seaded võimaldavad sul juhtida simulatsiooni “kiirust”.

Kuvamise kvaliteet (Disp.-Qual)

Kuidas kuvatakse eeltöödeldud simulatsioon Blenderi 3D-vaates (rippmenüü pealkirjaga *Viewport Display* (esitus vaates)) ja renderdamisel (rippmenüü pealkirjaga *Render Display* (esitus renderdamisel)):

- Geometry (geomeetria):: kasuta originaalgeomeetriat (enne simulatsiooni).
- Preview (eelvaade):: kasuta eelvaate võret.
- Final (lõplik):: Kasuta kõrglahutusega võret.

Kui eeltöötlemisandmeid ei leita, kuvatakse vaikimisi originaalvõre.

Pärast toimimispiirkonna eeltöötlemist kuvatakse see (harilikult) Blenderi aknas eelvaate võrena. Et näha toimimispiirkonna originaalsuurust ja ulatust, vali vasakust rippmenüüst Geometry (geomeetria).

Eeltötluse kataloog (Bake Directory)

VAJALIK! Kataloog ja failide eesliide, millega eeltööteldud pinnavõred salvestatakse. See on samane animatsiooni väljundseadetele, ainult faili valimine on natuke erinev: kui valid mõne eelnevalt loodud pindvõre (näiteks “untitled_OBcube_fluidsurface_final_0132.bobj.gz”), siis eesliiteks seatakse sel juhul automaatselt (“untitled_OBcube_”). Nii saab simulatsiooni teha erinevate seadetega ja see lubab kiireid muudatusi erinevate pinnaandmete seadete vahel.

Vaikeväärtus on “/tmp/”, mida sa ilmselt “ei” taha. Vali sobiv kataloogi nimi ja faili eesliide, nii et need failid salvestataks sobivasse kohta ja oleks nimetatud nii, et kaks erinevat vedeliku simulatsiooni ei kirjutaks üksteist üle (kui kavatsed määrata ainult kataloogi nime, näiteks ilma failinime eesliiteta, ära unusta lisada “/”).

Nupp “Bake” (eeltöötle)

Teostab tegeliku vedeliku simulatsiooni. Blender jätkab normaalselt tööd, kui välja arvata see, et akna üläääres, renderdamise rippmenüü kõrval näidatakse olekuriba. Vajutades kiirklahvi Esc või “x” olekuriba kõrval, saad simulatsiooni katkestada. Hiljem on iga kaadri jaoks valitud kataloogis kaks “.bobj.gz” faili (üks lõpliku (Final) kvaliteedi, üks eelvaate (Preview) kvaliteedi jaoks), pluss üks “.bvel.gz” (lõpliku (Final) kvaliteedi) fail.

Märkused...


Eeltöödeldud andmete vabastamine

“Bake” kataloogi sisu kustutamine on destruktiivne viis seda teha. Ole ettevaatlik, kui rohkem kui üks simulatsioon kasutab sama eeltötluse kataloogi (tee kindlaks, et neil on erinevad failinimed, muidu hakkavad nad üksteist üle kirjutama)!

Eeltötluste taaskasutamine

Eelnevalt salvestatud (eeltöödeldud) kataloogi ja failinime maski käsitsi sisestamine (või otsimine) vahetab vedeliku voolu ja võre moonutused nende vastu, mis olid selle vana eeltötluse ajal. Seega saad eeltöödeldud voogusid kasutada lihtsalt sellel väljal neile osutades.

Eeltöödeldud toimimispiirkonna valimine

Pärast toimimispiirkonna eeltöötlemist muutub see vedeliku võreks. Et toimimispiirkonda uuesti valida, nii et saaksid seda pärast tehtud muudatusi jälle eeltöödelda, mine mõnele kaadri ja selekteeri (RMB  vedeliku võre. Siis võid klõpsata uuesti nuppu BAKE (eeltöötle), et arvutada uuesti vedeliku voolamine selles toimimispiirkonnas.

Teised toimimispiirkonna valikud

Domain World (toimimispiirkond maailmas)

Viscosity (viskoossus)

Vedeliku “paksus” - tegelikult jõud, mida on vaja, et kindla pindalaga objekti kindla kiirusega sellest läbi liigutada. Võid sisestada väärtuse käsitsi või kasutada mõnda rippmenüüs olevatest algseadistustest (nagu mesi, õli või vesi). Käsitsi sisestamisel pane tähele, et reaalses maailmas mõõdetakse viskoossust (nn dünaamiline viskoossus) paskal-sekundis (Pa.s), või Poise'i ühikutes (P, võrdne 0.1 Pa.s, hääldatakse “*puaz*”, nimetatud prantslase Jean-Louis Poiseuille järgi, kes avastas “viskoossete vedelike laminaarse voolamise” seadused), ja harilikult sentipoise'ides (cP, võrdne 0.001 Pa.s, “*sentipuz*”). Teisalt kasutab Blender aga kinemaatilist viskoossust (mis on dünaamiline viskoossus Pa.s, jagatud tihedusega kg.m^{-3} , mille ühik on $\text{m}^2 \cdot \text{s}^{-1}$). Allpool olev tabel annab mõned näited vedelikest koos dünaamilise ja

kinemaatilise viskoossusega.

Käsitsi sisestatud väärtused määratakse ujukomaarvu ja astendajaga. Need ujukomade ja astendajate jaoks mõeldud väljad (teaduslik tähistus) lihtsustavad väga väikeste või väga suurte numbrite sisestamist. Vee viskoossus toatemperatuuril on 1.002 cP või 0.001002 Pa.s; vee tihedus on umbes 1000 kg.m⁻³, mis annab kinemaatiliseks viskoossuseks 0.00001002 m².s⁻¹ - seega sisestus peaks olema 1.002 korda 10 astmel miinus kuus (1.002×10⁻⁶ teaduslikus tähistuses). Tuline klaas ja sularaud on vedelikud, kuid väga paksud; nende jaoks peaksid sa kineetilise viskoossusena sisestama midagi sellist nagu 1.0×10⁰ (= 1.0) (mis vastab umbes väärtusele 1.0×10⁶ cP).

Pane tähele, et simulaator ei sobi mittevedelike ehk "mittevoolavate" materjalide jaoks. Seades viskoossuse väga suurtele väärtustele, saame jäiga keha käitumise, kuid see võib põhjustada ebastabiilsust.

Viskoossus varieerub

Blenderi vaikeväärtused on tüüpilised seda sorti vedelikele ja paistavad animeerituna "õiged". Ent mõnede vedelike tegelik viskoossus, eriti suhkrurikaste vedelike, näiteks šokolaadisiirupi ja mee viskoossus sõltub suuresti temperatuurist ja kontsentratsioonist. Õli viskoossus varieerub SAE märgistuse järgi. Klaas toatemperatuuril on põhiliselt tahke, kuid 1500 kraadi juures voolab (peaaegu) nagu vesi.

Blenderi viskoossuse ühikute teisendus

Vedelik	dünaamiline viskoossus (cP)	kinemaatiline viskoossus (Blender, m ² .s ⁻¹)
Vesi (20 °C)	1.002×10 ⁰ (1.002)	1.002×10 ⁻⁶ (0.00001002)
Õli SAE 50	5.0×10 ² (500)	5.0×10 ⁻⁵ (0.00005)
Mesi (20 °C)	1.0×10 ⁴ (10 000)	2.0×10 ⁻³ (0.002)
Šokolaadisiirup	3.0×10 ⁴ (30 000)	3.0×10 ⁻³ (0.003)
Ketšup	1.0×10 ⁵ (100 000)	1.0×10 ⁻¹ (0.1)
Sulaklaas	1.0×10 ¹⁵	1.0×10 ⁰ (1.0)

Realworld-size (suurus reaalses maailmas)

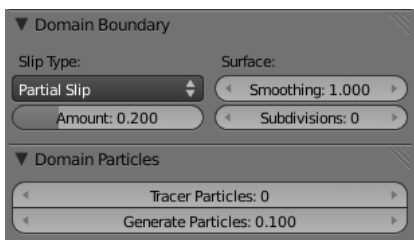
Toimimispiirkonna objekti suurus reaalses maailmas meetrites. Kui tahad luua tassi kohvi, võiks see olla 10 cm (0.1 meetrit), samas ujumisbassein võiks olla 10 m. Suurus seatakse siin toimimispiirkonna piirdkasti pikima külje järgi.

Gridlevel (ruudustiku tase)

Kui palju adaptiivseid ruudustiku tasemeid simulatsioonis kasutatakse - seades selle väärtuseks -1, on valik automaatne.

Compressibility (kokkusurutavus)

Kui sul probleeme suurte seisva vedeliku kogustega, millel on kõrge lahutus, siis võib aidata, kui seda numbrit vähendada (pane tähele, et see suurendab arvutusaega).



Teised vedeliku toimimispiirkonna valikud

Domain Boundary (toimimispiirkonna piirid)

Selles kastikeses on kõik libisemise ja pinna valikud.

Boundaries (piirid)

Boundary type (piiri tüüp)

See on sama nagu takistusobjektidel allpool, põhimõtteliselt seab see toimimispiirkonna kuus külge kas kleepuvaks, mittekleepuvaks või kusagile vahepeale (seda seatakse välja PartSlip Amount (osakeste libisemise määr) väärtusega).

Surface Smoothing (pinnasilumine)

Vedeliku pinnale rakendatava silumise määr. 1.0 on standardne, 0 on väljalülitatud, suuremad väärtused suurendavad silumist.

Subdivisions (tükeldustasemeid)

Lubab luua kõrgeresolutsioonilisi pindvõresid otse simulatsiooni käigus (vastupidiselt pinnatükeldamise töötlejale, mis teeb seda hiljem). Väärtus 1 tähendab, et pinnatükeldamist ei ole ja iga suurendamine annab juurde ühe pinnatükeldamise taseme iga vedeliku vokseli kohta. Tulemuseks saadav võre kasvab kiiresti mahukaks ja vajab palju kettaruumi. Ole ettevaatlik selle kombineerimisel suurte silumisväärtustega - see võib võre pinna genereerimisel viia pikkade arvutusaegadeni.

Generate SpeedVecs/Disable (Loo kiirusvektorid/lülita välja)

Kui seda nuppu klõpsata, siis kiirusvektoreid ei ekspordita. Seega vaikimisi luuakse kiirusvektorid ja salvestatakse kettale. Seda saab kasutada komposiitsõlmedega pildipõhise liikumishägu arvutamisel.

Domain Particles (toimimispiirkonna osakesed)

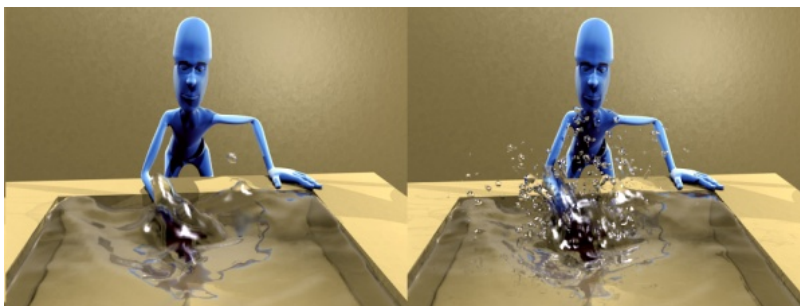
Siin saad vedeliku simulatsioonile lisada osakesi, et visuaalset efekti parandada.

Tracer Particles (jälgivad osakesed)

Simulatsiooni alguses vedelikku lisatavate järgivate osakeste arv. Et neid kuvada, loo teine objekt vedeliku tüübiga Particle (osake), mida on selgitatud allpool ja mis kasutab sama eeltöötuse kataloogi kui toimimispiirkond.

Generate Particles (loo osakesed)

Määrab, kui palju vedeliku osakesi luua (0=väljas, 1=tavaline, >1=rohkem). Et seda kasutada, peab pinnatükeldamise väärtus olema vähemalt 2.



Siin on näha osakeste efekti - vasakpoolne pilt on simuleeritud ilma ja parempoolne koos osakeste ning pinnatükeldusega.

Vedelik

Kõiki selle objekti osasid, mis asuvad seespool toimimispiirkonna piirdkasti, kasutatakse simulatsioonis vedelikuna. Kui lisad toimimispiirkonnale rohkem kui ühe vedeliku, siis hetkel ei peaks nad lõikuma. Samuti veendu, et pinnanormaalid osutaksid väljapoole. Erinevalt toimimispiirkonna objektidele, kasutatakse vedelikuobjektide puhul tegelikku võre geomeetria.

Volume init (mahu algseadistamise tüüp)

- Volume Määrab objekti sisemise osa vedelikuna. See töötab ainult suletud objektidega.
- Shell (kest) määrab vedelikuna ainult õhukese kihi võre kõikidel külgedel. See töötab ka lahtistel võredest.
- Both (mõlemad) kombineerib mahu ja kesta - sel juhul peaks võre olema samuti suletud. Vaata allpool asuvat pilti.



Erinevate mahu algseadistamise tüüpide näide: maht (Volume), kest (Shell) ja mõlemad (Both). Pane tähele, et kest on tavaliselt natuke suurem kui maht.

Initial velocity (algkiirus)

Vedeliku kiirus simulatsiooni alguses, meetrites sekundi kohta.



Pinnanormaalide suunal on suur vahe!

Blender kasutab pinnanormaalide suunda, et otsustada, mis on vedelikuobjekti "sees" ja mis on "väljas". Sa tahad, et kõik normaalid osutaksid *väljapoole* (vajuta muutmisrežiimis kiirklahvi Ctrl+N või vajuta Space ja vali Edit → Normals → Calculate Outside (muuda->normaalid->arvuta väljapoole)). Kui normaalid osutavad vales suunas, saad sa "hiiglasliku veeuputuse", sest Blender arvab, et objekti maht on väljaspool selle võre! See toimub sõltumata mahu tüübi (Volume init) tüübi seadetest.

Takistus

Seda objekti kasutatakse simulatsioonis takistusena. Vedelikobjekt ja takistus objekt ei tohiks hetkel üksteist läbistada. Nagu vedelikuobjektide puhul, kasutatakse ka takistuste puhul tegelikku võre geomeetria. Mahuga objektide puhul veendu, et takistuse normaalid oleksid arvutatud õigesti ja näitaksid õiges suunas (kasuta muutmisrežiimis võretööriistade (Mesh Tools) paneelil muutmiskontekstis Editing nuppu Flip Normal (pööra normaalid ümber) (F9)), eriti kui kasutad pööramisvahendiga tehtud mahutit. Kui võre ei ole animeeritud, on enne eeltöötlust hea mõte rakendada pinnatükeldajat (SubSurf).

Volume init (mahu algväärtustamise tüüp)

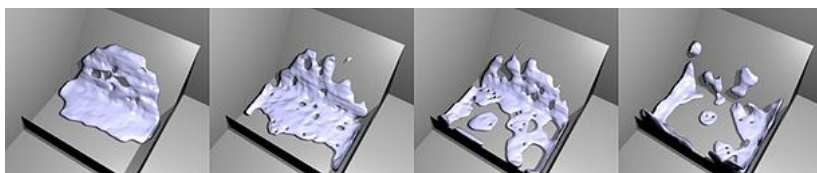
Sama mis ülalkirjeldatud vedelikuobjekti puhul.

Boundary type (piiri tüüp) (vaata pilti allpool)

Määrab takistuse pinna kleepuvuse, nimetatakse ka "pinna adhesiooniks". Reaalses maailmas sõltub pinna adhesioon vedelikust ja pinna teralisusest või hõõrdumise/adhesiooni/imevuse väärtustest.

- Noslip (mittelibisev) põhjustab vedeliku kleepumise takistuse külge (nullkiirus).
- Free (vaba libisemine) lubab liikumist piki takistuse pinda (ainult kiirus normaali suunas on null).
- Part (osaline libisemine) segab mõlemad tüübid, kus väärtus 0 on peamiselt mittelibisev ja 1 vabaltlibisev.

Pane tähele, et kui võre liigub, koheldakse seda automaatselt mittelibisevana.



Näide erinevate piiri tüüpide mõjust, kui tilk kukub kaldpinna. Vasakult paremale: mittelibisev, osaline libisemine 0.3, osaline libisemine 0.7 ja vaba libisemine.

Animated Mesh/Export (animeeritud võre/eksport)

Klõpsa seda nuppu, kui võre on animeeritud (näiteks moonutatud skeleti, võtmevormi või sõrestiku poolt). Pane tähele, et see on märgatavalt aeglasem ja ei ole vajalik, kui võre on animeeritud ainult positsiooni või pöörlamise IPOdega (st ainult *objekti* teisendustega).

PartSlip Amount (osalise libisemise määr)

Eelpool kirjeldatud mittelibisemise ja vabaltlibisemise segamise hulk.



Blender 2.5 toetab nüüd liikuvaid takistusi!

Kui takistus liigub, kohtleb Blender seda automaatselt mittelibisevana (kleepuvana). Kui tahad, et vedelik pritsiks liikuvast objektist eemale, pane sinna, kus vedelik liikuva objektiga kokku puutuks, läbipaistev tasapind, mis on täpselt sama kujuga kui objekt, kuid veidike objektist ettepoole, vedeliku ja objekti vahele. Kui objekt tuhiseb mööda ja vedelik pritsib, siis pritsimine puutub tegelikult vastu läbipaistvat pinda ja libiseb eemale ning objekt jätkab oma teed

Impact Factor (mõjufaktor)

Vedeliku mahu suurenemise/vähendamise korrigeerimine liikuva objektiga kokkupõrkel. Kui objekt ei liigu, ei ole sellel seadel mingit efekti. Kui ta aga liigub ja vedelik temaga kokku põrkab, siis negatiivne väärtus vähendab vedeliku mahtu toimumispiirkonnas ja positiivne suurendab seda. Vahemik on -2.0 kuni 10.0.

Sissevool

See objekt lisab simulatsioonile vedeliku (umbes nagu veekraan).

Volume init (mahu algväärtustamise tüüp)

Sama mis ülalkirjeldatud vedelikuobjekti puhul.

Inflow velocity (sissevoolu kiirus)

Kiirus, millega objekti sees vedelikku luuakse.

Local Coords/Enable (kohalik koordinaadistik/lülita sisse)

Kasuta sissevoolul kohalikke koordinaate. See on kasulik, kui sissevoolu objekt liigub või pöördub, sest siis sissevoolu juga järgib/kopeerib seda liikumist. Kui see on välja lülitatud, siis sissevoolu asukoht ja suund ei muutu.

Animated Mesh/Export (animeeritud võre/eksporti)

Sama mis ülalkirjeldatud takistusobjekti puhul.

Väljavool

Vedelik, mis selle objekti piirkonda satub, kustutatakse (nagu äravool või must auk). Seda kasutatakse kombineerituna koos sissevooluga, et vältida toimumispiirkonna vedelikuga täitumist. Kui see on sisse lülitatud, näeb ta välja nagu tornaado (vesipüks) või tolmuimeja ja see osa, kust vesi kaob, järgib objekti, kui see ringi liigub.

Volume init (mahu algväärtustamise tüüp)

Sama mis ülalkirjeldatud vedelikuobjekti puhul.

Animated Mesh/Export (animeeritud võre/eksporti)

Sama mis ülalkirjeldatud takistusobjekti puhul.

Osake

Seda tüüpi kasutatakse simulatsiooni käigus loodud osakeste kuvamiseks. Praegu toetatakse ainult osakesi, mis ujuvad koos vedelikuga. Pane tähele, et sellel objektil võib olla suvaline kuju, asukoht või tüüp: kui vajutada osakeste nuppu, luuakse osakeste süsteem koos vedeliku simulatsiooni osakestega õigesse kohta. Kui algset objekti liigutada, võib osutada vajalikuks kustutada osakeste süsteem, välja lülitada vedelikusimulatsiooni osakesed ja siis need uuesti sisse lülitada. Teised osakeste jõud või seaded ei mõjuta vedelikusimulatsiooni osakesi.

Particle type (osakeste tüüp)

Drops (tilgad)

Vedeliku pinna pritsimisel tekivad tilgad, mis on laiali külvatud nagu madala pindpinevusega puhta vee puhul.

Floats (pritsmed)

Vedeliku pindpinevus on kõrgem ja vedelik raskem, nagu külm merevesi ja supp. Eraldunud osad on ebakorrapärasemad ja langevad pinnale tagasi kiiremini kui tilgad (Drops), nagu suure pindpinevuse puhul.

Tracer (jälgija)

Piisad järgivad seda kohta, kus vee pind on olnud, nagu udu, mis ripub eelmiste vedelike tasemete kohal. Kasuta seda, et näha, kus vedeliku tase on olnud.

Size Influence (mõju suurusele)

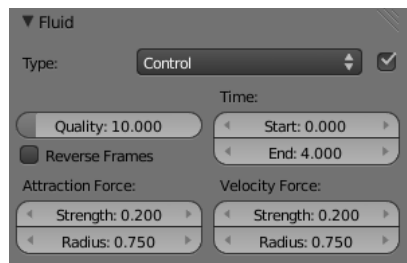
Osakesed võivad olla erineva suurusega, kui selle seade väärtus on 0, siis on nad kõik sama suurusega.

Alpha Influence (mõju katvusele)

Kui selle seade väärtus on >0, muudetakse osakeste katvuse (alfa) väärtust vastavalt nende suurusele.

Path (eeltöötuse kataloog)

Simulatsiooni tulemus, kust laaditakse osakesed. Siin peaks harilikult olema sama väärtus, mis vedeliku toimimispiirkonna objektile (näiteks kopeeri see kiirklahvidega CtrlC, CtrlV).

Juhtimine

Vedeliku juhtimise seaded.

Quality (kvaliteet)

Kõrgema kvaliteedi tulemusena luuakse juhtobjekti jaoks rohkem juhtosakesi.

Reverse Frames (tagurpidi kaadrid)

Juhtosakeste liikumine muudetakse vastupidiseks.

Time (aeg)

Saad määrata alguse ja lõpu aja, mille jooksul vedeliku juhtobjekt on aktiivne.

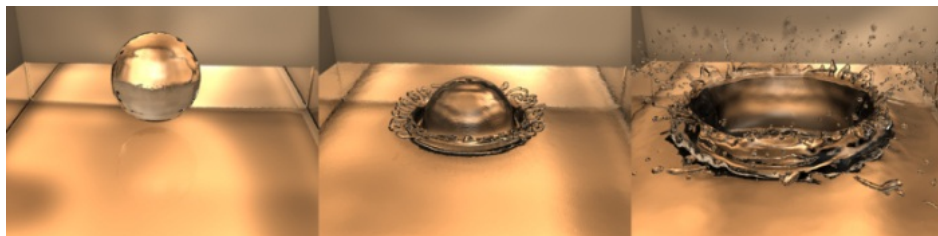
Attraction force (külgetõmbejõud)

Külgetõmbejõud määrab, millist jõudu vedeliku juhtobjekt emiteerib. Positiivne jõud tõmbab vedelikku ligi, negatiivne välidib.

Velocity force (kiiruse jõud)

Kui vedeliku juhtobjekt liigub, siis tema kiirus avaldab ka vedelikule jõudu.

Vaata täpsemat informatsiooni [nendest väljalaskemärkustest](#).



Veel üks näide kukkuvu tilga animatsioonist, mis on simuleeritud Blenderi ja renderdatud Yafray abil.

Vedeliku omaduste muutuste animeerimine

Vedeliku toimimispiirkonna objekti jaoks on olemas uut tüüpi IPO-kõver, FluidSim (vedelikusimulatsioon). Erinevalt teistest animeeritavatest väärtustest Blenderis, ei saa FluidSim IPOle võtmekaadrit määrata ainult kiirklahvi abil, vaid sa pead IPO aknas väärtused käsitsi sisestama. Et võtmekaadrit määrata, pead sa IPO aknas valima omaduse, mida soovid animeerida, ja klõpsama Ctrl LMB, et seada võtmekaader IPO aknas soovitud kohta.

**Sisesta omadused**

Pane tähele, et ei pea olema täpne, kuhu sa klõpsad; soovitame pärast kontrollpunkti seadmist avada teisenduse omaduste paneeli Transform Properties (N), ümardada X-väärtus täisarvulise kaadri numbrini ja siis määrata Y-väärtus, nii nagu soovid.

Vedeliku toimimispiirkonnal on mitu kanalit, mis juhivad vedelikku ajas:

Fac-Visc (viskoossuse koefitsient)

Vedeliku viskoossuse koefitsiendi korrutav parameeter. See tuleb seada enne eeltöötlust ja muudab aja jooksul vedeliku viskoossust, nii saad näiteks muuta vee õliks!

Fac-Tim (ajakoeftsient)

Muudab simulatsiooni kiirust; nii nagu kiiruse seade videorektoris suurendab või vähendab video kiirust, suurendab või vähendab see kõver vedeliku liikumise kiirust kaadrite järjestuses. Kui Fac-Tim väärtus on väiksem või võrdne nulliga, siis aeg (ja vedelik) seisab paigal; vedelik tardub. Vahemikus 0.0 kuni 1.0 vahel voolab vedelik aeglasemalt ja paistab tihedam. 1.0 on normaalne vedeliku liikumine ja väärtused, mis on suuremad kui 1.0, panevad vedeliku voolama kiiremini ja võimalik, et paistma õhemana.

GravX/GravY/GravZ (gravitatsiooni X/Y/Z)

XYZ-vektor gravitatsiooni muutusteks ehk vedeliku enda inerts (kujutle kohvijoomist ringrajaauto roolis, espresso lonksamist kiirteel või taimede kastmist kosmosesüstikus). Tehes muudatusi nendes kõverates, paned sa vedeliku väliste jõudude toimele loksuma.

The Fluid (vedelik), Obstacle (takistus), Inflow (sissevool), Outflow (väljavool) ja Particle (osake) tüüpi objektid võivad kasutada järgmisi kanaleid:

VelX/VelY/VelZ (kiirus X/Y/Z)

Vedeliku purskumist alavoolikust saab simuleeruda nende kõverate abil, imiteerimaks muutusi surves ja/või suunas. Seda saab näiteks kasutada ka tuule mõju simuleerimiseks veejoale.

Active (aktiivne)

Kui aktiivsus läheb väärtuselt 0.0 millekski suuremaks kui 0 (nagu näiteks 0.1 ja 1.0 vahel), hakkab vastav objekt (määratud kui Inflow (sissevool) või Outflow (väljavool) jne) jälle toimima. Langedes alla 0.0-ni ja tõustes siis mingil hetkel uuesti üles, taastub tema mõju ja sellega kaasnev vedeliku simulatsioon. Kasuta seda tilkumise või mõne muu katkendliku sissevoolu korral. See aktiivne olek töötab ka objektidel, mis on määratud kui Outflow (väljavool) ja Obstacle (takistus) - nii saad sa (näiteks) simuleerida äravoolule korgi ette panemist.

Samuti saad määrata juhtobjektide Control jõu seadeid:

AttrForceStr, AttrForceRa

Need kõverad juhivad külgetõmbejõu seadete väärtusi.

VelForceStr, VelForceRa

Need kõverad juhivad kiiruse jõu seadete väärtusi.

Tehnilised detailid



"My cup runneth over", loodud Blenderi ja Yafray abil.

Vedelikusimulatsioon võtab palju aega - mida paremini sa selle tööd mõistad, seda lihtsam on sul hinnata, kui realistlik tulemus saab olema. Algoritm, mida Blender vedeliku simulatsioonis kasutab, on *Boltzmanni meetod võrel* (LBM); teised vedeliku algoritmid on *Navier-Stokesi* (NS) lahendajad ja *Silutud osakeste hüdrodünaamika* (SPH) meetodid. LBM on kuskil nende kahe vahepeal.

Üldiselt on praegustel arvutitel väga raske simuleerida isegi 1-meetrist veetünni. Et simuleerida lainet murdmas läbi linna, on sul ilmselt vaja üht kõige kallimatest superarvutitest, mis saada on, ja ka see ei pruugi ikkagi töötada õigesti, ükskõik millist neist kolmest algoritmist sa kasutad. Seetõttu pead sa sellise efekti saavutamiseks, mida sa tegelikult soovid, võtma abiks strateegia, mis sarnaneb "analoogfilmides" palju aastaid kasutatud meetodiga: "võltsi seda!"

Hea vedelikusimulatsioon on *väga tähtis*, kuid mitte *ainuke* osa, et saavutada rahuldav pilt. Lase Blenderil teha arvutamise must töö vedeliku baassimulatsiooni arvutamisel, seejärel muuda see tõetruuks, lisades valitud detaile, mis vastavad vaataja ootusele "reaalsele veekeerise osas, mille sa oled loonud".

Näiteks saad sa teeselda, et sul on linnas hiidlaine, ehitades *väiksema* mudeli, modelleerides *väikese* laine linna mudelisse väga

kõrge lahutusega ja loota, et keegi ei märka erinevust 100 m ja 1 m laine vahel (ei märkagi). Lisa lainefrondile tekstuur müra ja pilvedega, mis mõjutavad värvust. Lisa rohkesti suitsu (udu) emiteerijaid erinevatele pindadele, mida laine tabab, ajastades need emiteerima kokkupõrke hetkel kokkupõrke ja pinna suunas. Animeeri autod ja prügi (ja uppuvad inimesed...) lainefrondile hulpima ja ujuma, kasutades eeltöödeldud võret. Kasuta laine harjal rida udu emiteerijaid, et simuleerida veeudu, mis laine harjalt õhku paiskub. Mõtle täpselt järele, kuhu sa tahad paigutada kaamera, kas kasutada suumobjektiivi või lainurka ja nii edasi (kas vaataja "vaatab vaestele näitlejatele ülalt alla" või "upub koos nendega"?). Sedasorti tähelepanu muudele detailidele peale vedelikusimulatsiooni enda teebki võttest selle, mis ta olema peab.

Blenderi LBM-i lahendajale teevad arvutamise raskemaks järgmised asjad:

- Suured toimimispiirkonnad.
- Pikad kestused.
- Madalad viskoossused.
- Suured kiirused.

Vee viskoossus on niigi madal ja eriti väikse lahutuse korral ei tabata vee turbulentsi päris õigesti. Kui vaatad lähemalt, siis enamik vedelikusimulatsioone ei paista praegu väga reaalse vee moodi. Üldiselt, ära tugine liiga palju füüsilistele seadetele (nagu toimimispiirkonna füüsiline suurus või animatsiooni pikkus sekundites). Parema ürita saavutada üldine meeleolu madala lahutusega ja siis suurenda seda nii palju kui võimalik või soovitud.

Vihjed

- Ära ehmata, sa saad pärast simulatsiooni terve portsu võre (.bobj.gz) faile. Üks komplekt eelvaate ja teine lõpliku renderduse jaoks. Igas komplektis on üks .gz fail iga animatsiooni kaadri jaoks. Iga fail sisaldab simulatsiooni tulemust - seega sa vajad neid.
- Praegusel hetkel ei kustutata neid faile automaatselt, seega on hea mõte luua simulatsiooni tulemuste säilitamiseks spetsiaalne kataloog. Vedelikusimulatsiooni tegemine on nagu nupu ANIM klõpsamine - praegu pead sa vedeliku pinna võred ise kuhugi kataloogi panema. Kui soovid simulatsiooni kasutamise lõpetada, võid lihtsalt kustutada kõik *fluid*.bobj.gz failid.
- Enne kõrge lahutusega simulatsiooni jooksutamist, mis võib võtta tunde, kontrolli üldist ajastust, jooksutades esmalt madala resolutsiooniga.
- Vedelikuobjektid peavad olema täielikult toimimispiirkonna objekti piirdkasti sees. Kui ei ole, ei pruugi eeltöötlus õigesti toimida. Vedelikobjektid ja takistused võivad olla keeruka geomeetriaga võred. Väga õhukesed objektid ei pruugi simulatsioonis näha olla, kui valitud lahutus on liiga jäme (lahutuse suurendamine võib selle probleemi lahendada).
- Pane tähele, et selliseid vedelikusimulatsiooni parameetreid nagu sissevoolu kiirus saab animeerida Fluidsim IPOdega (vaata ülaltpoolt).
- Ära ürita teha keerukat stseeni ühe korraga. Blenderil on mitmekülgne komposiitor, mida saad kasutada mitme animatsiooni kombineerimisel.

Näiteks selleks, et teha animatsioon, mis näitab kahte erinevat vedeliku voolu, hoides samal ajal toimimispiirkonda väiksena, renderda üks .avi-fail, kasutades ühte voolu. Siis liiguta toimimispiirkonda ja renderda teine .avi-fail teise vooluga, kasutades alfakanalit (eraldi mustvalge .avi?). Siis komposiidi mõlemad .avi-failid, kasutades komposiitori liitmiskompositsiooni. Kolmas .avi on harilikult suitsu ja udu jaoks ning pannakse ka kõige peale. Lisa veel udule vihmakiht ja pritsmed ja sul on päris torm möllamas! Ja siis välgulöögid, ringilendlev praht, kõik eraldi animatsioonid, komposiititud kokku tõeliselt suurepäraseks tulemuseks.

- Kui sul on probleeme või miski ei tööta nii nagu vaja - anna mulle teada: saada .blend-fail ja probleemi kirjeldus `nils at thurey dot de`. Vaata selle viki lehekülgi ja [blenderartists-foorumit](#) enne maili saatmist!

Piirangud ja lahendused

- Ühe Blenderi faili kohta üks toimimispiirkond (versioon 2.42), aga sul võib olla mitu vedelikuobjekti.

Lahendus: tõsta eelvaateks toimimispiirkonda ringi, et ta hõlmaks iga vedelikuvoolu osa, lõpuks suurenda toimimispiirkonda nii, et ta sisaldaks kõiki vedelikuobjekte (kuid arvutusaeg tuleb pikem). Õigupoolest on see eelis, sest varieerides toimimispiirkonna suurust ja asukohta, saad sul valida, kui palju arvutusaega kasutatakse.

- Kui süsteem paistab valesti minevat, tee kindlaks, et kõik normaaliid on õiged (mine muutmisrežiimi, vali kõik ja arvuta normaaliid uuesti).
- Hetkel on probleeme nullgravitatsiooniga simulatsiooniga - vali lihtsalt väga väike gravitatsiooni suurus, et seda parandada.
- Kui objekt on määratud kui maht (Volume), peab see olema suletud ja tal peab olema sisemine külg (tasapind ei toimi). Et kasutada tasapindu, lülita tüübiks kest (Shell) või suru tasapind välja.
- Blender hangub pärast nupu BAKE klõpsamist. Vajutades Esc, hakkab ta mõne aja pärast uuesti tööle - see võib juhtuda, kui lahutus on liiga kõrge ja mälu vahetatakse kõvakettale, tehes kõik väga aeglaseks. Sel juhul peaks aitama lahutuse vähendamine.
- Pärast nupu BAKE klõpsamist jookseb Blender kokku - see võib juhtuda, kui lahutus on tõeliselt kõrge ja hõivatud on rohkem kui 2 GB mälu, põhjustades Blenderi kokkujooksmise. Vähenda lahutust. Pea meeles, et mitmetel operatsioonisüsteemidel on piiratud see, kui palju mälu ühele "protsessile", nagu näiteks Blenderile eraldatakse, isegi kui "masinal" on mälu sellest rohkem. Nõme...
- Võred peaksid olema suletud - seega, kui näiteks vedelikuobjekti mõned osad ei toimi simulatsioonis vedelikuna, siis kontrolli, kas kõik ühendatud tippude grupid on suletud võred. Kahjuks ei ole Suzanne'i (ahvi) võre Blenderis suletud võre (silmad on eraldi).

- Kui vedeliku simulatsioon lõpetab veateatega (näiteks "init has failed"), tee kindlaks, et toimimispiirkonna objekti seaded on õiged, näiteks taastades algseaded.
- Et importida üksikut vedeliku võret, võid kasutada seda skripti: [.obj-Import-Script](#).
- Võib juhtuda, et sa ei saa eeltöödelda vedelikku, mis võtab mälu rohkem kui 1 GB, isegi mitte siis, kui Blender on kompileeritud LargeAddressAware'i seadetega - see võib olla praeguse vedelikumootori piirang.
- Pane tähele, et esimene kaader võib võtta mälu ainult mõnisada MB, kuid hilisemad lähevad juba üle ühe GB, mis võib olla põhjuseks, miks eeltöötlus ebaõnnestub. Kui nii, siis proovi eeltöödelda üks kaader keskelt või lõpust täislahutusega, et näha, kas see töötab.
- Mälu kasutus kahekordistub, kui sead pinnatükeldamise 1st 2le.
- "Osakeste genereerimine" suurendab samuti mälu kasutamist, kuna need suurendavad pinda ja keerukust. Tavalised vedelikusimulatsiooni osakesed võtavad ilmselt vähem mälu.

TEHA: kontrolli neid linke...(Et nad oleks sobivad Blender 2.5.ga

Vaata ka

- [Õppetükk 1: Väga algeline sissejuhatus](#)
- [Õppetükk 2: Järgmine samm](#)
- [Õppetüki 1&2 kasutajaliidese muudatused uute versioonide jaoks](#)
- [Veel üks BSoD vedeliku õppetükk](#)
- [Arendusdokumentatsioon \(lahendus, vajalikud komponendid jms\)](#)

Väline dokumentatsioon

- [An Introduction to Fluid Simulations in Blender \(video\) \(Blendernation link\)](#) - Õpi, kuidas luua Blenderis vedelikusimulatsiooni koos takistusega.
- [Fluid Simulator Tutorial \(video\) \(Blendernation link\)](#) - Väga lihtsasti arusaadav algajate videoõppematerjal vedelikusimulatsioonist. Käsitleb ka mõningaid enamlevinud vigu.
- [Guide on Blender Fluid Simulator's Parameters \(Blendernation link\)](#)

Tänu sõnad

Vedelikusimulaatori integreerimine tehti Google Summer-of-Code projektina. Rohkem infot lahendaja kohta leiab www.ntoken.com. Need animatsioonid on tehtud enne lahendaja integreerimist Blenderisse: [Adaptive Grids](#), [Interactive Animations](#). Tänud Chris Wantile Blender-SoC projektide organiseerimise eest ja Jonathan Merritile selle mentoriks olemise eest! Ja muidugi tänud Google'ile kogu asja alustamise eest... SoC edenemine ja uuendused: [SoC-Blenderfluid-Blog at PlanetSoC](#).

Lahendaja ise on arendatud Nils Thuerey poolt ja järgnevate inimeste abiga ning järelevalve all: U. Ruede, T. Pohl, C. Koerner, M. Thies, M. Oechsner ja T. Hofmann, Arvutiteaduse osakonnas 10 (System Simulation, LSS) Erlangenis Saksamaal.

<http://www10.informatik.uni-erlangen.de/~sinithue/img/lsslogo.png>

<http://www10.informatik.uni-erlangen.de/~sinithue/img/unierlangenlogo.png>

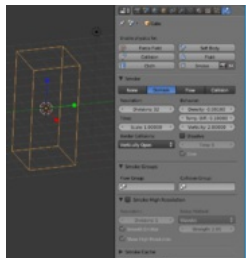
Suitsu simulatsioon

Blenderi uus suitsusimulatsioon põhineb artiklil [turb.php/Wavelet Turbulence for Fluid Simulation](http://turb.php/Wavelet_Turbulence_for_Fluid_Simulation) ja vastaval näidiskoodil, ning on Blenderisse porditud Daniel Genrichi poolt ja töökorras hoiab seda praegu Janne Karhu.

Simulaator põhineb mahulisel vedelikupõhisel mudelil, mille lõpptulemuseks on vokselite võre. Blenderis visualiseeritakse vokselite andmed interaktiivselt, kasutades OpenGL-i varjutamist ja renderdatakse vokseliandmete (*Voxel Data*) tekstuuri abil. Blenderi **suitsusimulatsioon** mähib vokslid ümber olemasolevate [osakeste](#). See vajab osakesi emiteerivat objekti ja 'toimimispiirkonna' objekti, mille sees suits renderdatakse.

See osa dokumentatsioonist vastab versioonile 2.58

Suitsusimulatsioon on samane vedelikusimulatsioonile. Simulatsiooni jaoks on vaja toimimispiirkonna objekti ja voolu objekti.



Suitsu toimimispiirkonna objekt

Nagu vedeliku simulatsioonil, on enamuse seadeid näha, kui toimimispiirkonna objekt on valitud.

Töö käik

- Loo või määra [toimimispiirkonna](#) (*Domain*) objekt või võre, mis määrab simulatsiooni piiri.
- Loo või määra [voolu](#) (*Flow*) objekt, mis määrab, kust suitsu väljutatakse.
- Määra [kokkupõrke](#) (*Collision*) objektid, mis stseenis suitsu mõjutavad.

Toimimispiirkond (*Domain*)

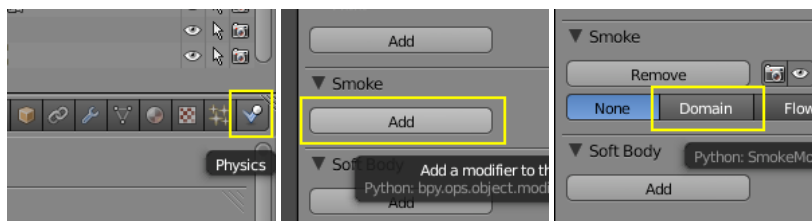
Loo toimimispiirkond

Enne kui saad oma stseenile suitsu lisada, pead määrama ala, kus suitsu simulatsioon toimub. Blenderi füüsikas nimetatakse seda toimimispiirkonnaks (*Domain*). Hea idee on selleks valida kuup, kuna sa saad hiljem selle skaleerida kaamera vaateväljale vastavaks. Meie teeme lihtsal vaikekuubi suuremaks, vajutades kiirklahvi S ja lohistades hiirega.

Ära muuda toimimispiirkonna tippu!

Kui tahad suuremat toimimispiirkonda, siis skaleeri objekti. Selle muutmise muutmisrežiimis põhjustab suitsu ilmutamise renderdamise ajal rohkem kui ühes kohas, nagu korduva tekstuuri puhul.

Veendu, et oled objektirežiimis ja mine füüsika (*Physics*) kaardile. Lisa suits (*Smoke*) ja vali nupp tekstiga 'Domain' (toimimispiirkond). Praeguseks on see kõik, me pöördume väljailmunud uute seadete juurde hiljem tagasi.



Valikud

Resolution (lahutus)

Kui detailne suits on. Lahutusel 32 võtab eeltöötlus mõne sekundi, samas lahutusel 100 võib selleks enamikul PC-del kuluda kuni pool tundi.

Time Scale (ajaskaala)

Mõjutab simulatsiooni esitamise kiirust.

Border Collisions (servakokkupõrked)

Vertically Open (vertikaalselt avatud)

Suits kaob, kui ta puutub vastu toimimispiirkonna põhja või lage.

Open (avatud)

Suits kaob, kui läbib toimimispiirkonna piiri.

Collide All (põrka kõigiga)

Toimimispiirkonna piire koheldakse kui põrkeobjekti, suits põrkub ja jääb toimimispiirkonna sisse.

Temperature and Density (temperatuur ja tihedus)

Kui palju tihedus ja temperatuur suitsu liikumist mõjutavad. Kõrgemad väärtused teevad kiiremini tõusva suitsu.

Vorticity (keerised)

Määrab, kui turbulentne/keerduv või keerlev suits on.

Dissolve (hajumine)

Laseb suitsul aja jooksul hajuda.

Time (aeg)

Suitsu hajumise kiirus.

Slow (aeglane)

Kasutab aja (*Time*) väärtuse asemel väärtust $1/Time$, tehes suitsu hajumise aeglasemaks.

Suitsugrupid**Suits kõrge lahutusega**

Kõrge lahutuse (*High Resolution*) valik lubab sul simuleerida madala resolutsiooniga ja seejärel kasutab müratehnikaid, et lahutust suurendada, ilma seda tegelikult läbi arvutamata. See võimaldab animeerijal valmistada kiiresti ette madala lahutusega simulatsioon ja lisada hiljem detaile, muutmata üldist vedeliku/gaasi liikumist.

Selle jaoks on võimalikud erinevad meetodid, kaasa arvatud vaikemeetod Wavelet, mis realiseerib '[turb.php|Wavelet Vedelikusimulatsiooni turbulentsi](#)' meetodi

Resolution/Divisions (lahutus/jagamisi)

Suurendab suitsu lahutust müra abil selle teguri võrra.

Smooth Emitter (sile emiteerija)

Silub emiteeritud suitsu, et vältida suitsu kandilisust.

Show High Resolution (näita kõrglahutust)

Näitab kõrget lahutust, kasutades võimendust.

Noise Method (müra meetod)

Wavelet

FFT (Fourier)

Strength (tugevus)

Müra tugevus.

Suitsuvälja kaalud

Määrab, kui palju erinevad jõud ja jõuväljad suitsu mõjutavad.

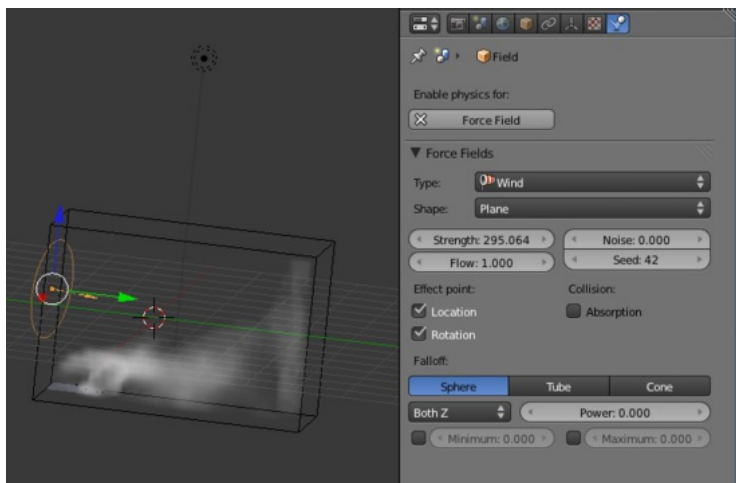
Gravity (gravitatsioon)

Kui palju gravitatsioon suitsu mõjutab.

All (kõik)

Muudab kõikide jõuväljade üldist mõju.

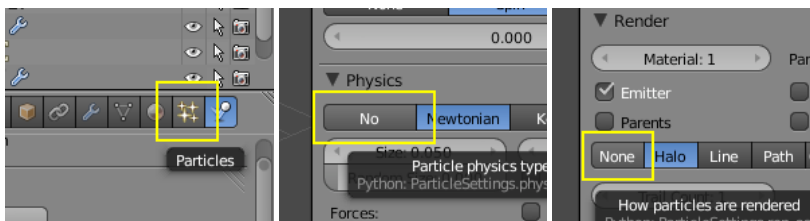
Ülejäänud seaded määravad, kui palju erinevad jõuväljad suitsu mõjutavad.

**Sissevool (Inflow)****Loo voolu objekt**

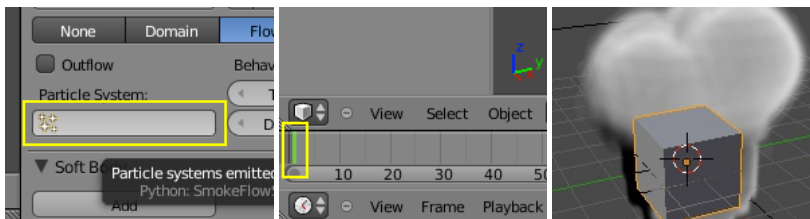
Nüüd, kui me oleme määranud mahu, mille sees suits hakkab olema, lisame objekti, millest suitsu hakatakse emiteerima. Lisa teine kuup ⇨ ShiftA ⇨ Mesh ⇨ Cube (võre->kuup) ja veendu, et see oleks toimimispiirkonna kuubi sees; valitud peab olema 3D-vaade).

Olles muutmisrežiimis, mine füüsika kaardile ja lisa väikesele kuubile samuti suits. Seekord vali vool (Flow).

Suitsu ei emiteerita mitte objektist endast, vaid osakestest, mida objekt emiteerib. Seega, peame looma osakeste süsteemi. Väike kuup ikka veel valitud, mine osakeste kaardile. Lisa uus osakeste süsteem ja lülita välja füüsika, sest me soovime, et suits emiteeritaks liikumatust kohast. Samuti ei taha me näha osakesi, seetõttu lülita välja ka renderdamine.



Nüüd mine tagasi füüsika kaardile ja vali osakeste süsteem suitsu alajaotuses. Seal peaks olema süsteemide nimekiri, kust saab hetkel valida ainult ühe süsteemi, mille nimi on 'ParticleSystem', sest me ei muutnud nime. Nüüd saad kerida mööda ajatelge, et näha kuubikust suitsu tõusmas. Teine võimalus suitsu eelvaateks on animatsiooni mängimine kiirklahviga AltA (peata mängimine sama klahviga).



Seaded

Outflow (väljavool)

Kustutab simulatsioonist suitsu.

Particle System (osakeste süsteem)

Sellest objektist emiteeritud osakeste süsteem.

Initial Velocity (algkiirus)

Suits saab oma kiiruse emiteerija osakestelt.

Multiplier (kordaja)

Kordaja suitsule edasiantava kiiruse seadmiseks.

Algväärtused

Absolute Density (absoluutne tihedus)

Lubab emiteerija alas ainult etteantud tihedust.

Density (tihedus)

Algne tiheduse väärtus.

Temp. Diff. (temperatuuride erinevus)

Temperatuuri erinevus võrreldes ümbritseva temperatuuriga.

Kokkupõrked

Suits võib põrkuda teiste objektidega, kui kasutada tema 'Põrke' (*Collision*) valikut. Hetkel on toetatud ainult põrked staatiliste objektidega.

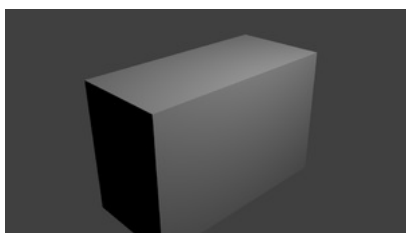
Jõud

Töötavad ka Blenderi jõuväljad (nagu tuul ja keerisväljad), mis muudavad suitsu simulatsiooni samamoodi nagu teisi füüsikalisi süsteeme, näiteks osakesi.

Suitsu renderdamine

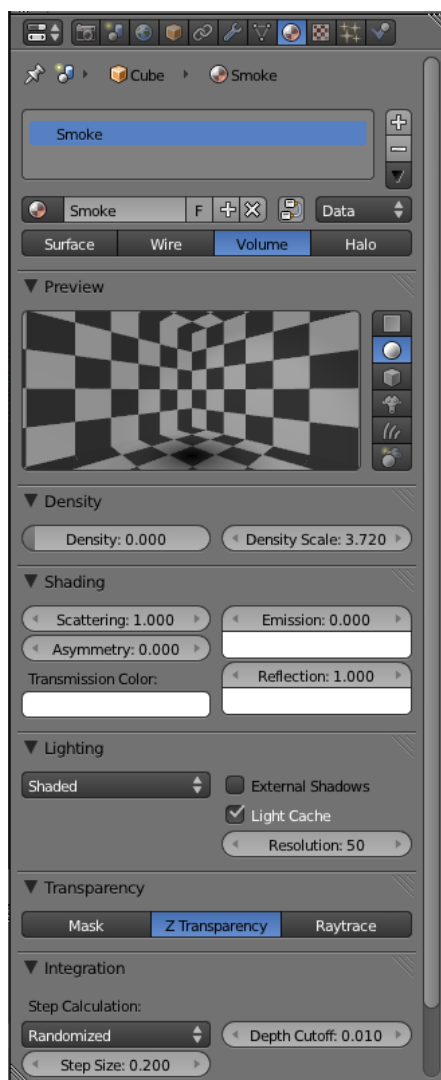
Loo materjal

Suitsu simuleerimine on lihtne, renderdamine samas aga mitte.

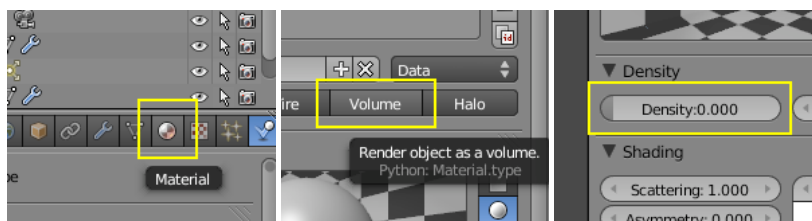


Praegu renderdamine annab tulemuseks lihtsalt suure risttahuka (nagu pildil, F12) või kokkujooksmise (animatsioon, CtrlF12).

. Materjal peab olema mahuline materjal tihedusega (*Density*) 0 ja suure tiheduse skaala (*Density Scale*) väärtusega.

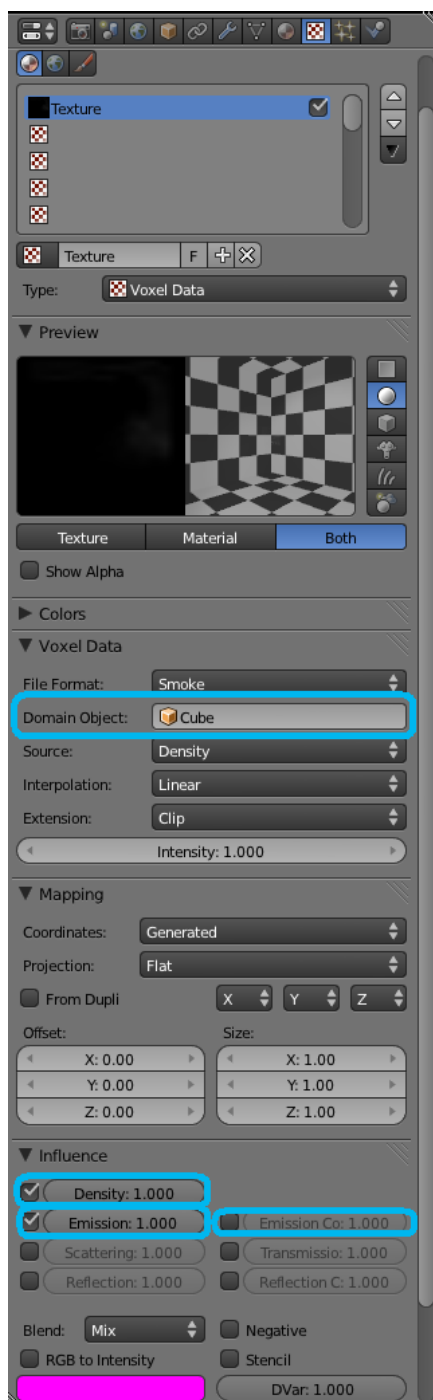


Esimese probleemi saab kergesti lahendada, kui määrata toimimispiirkonna kuubi jaoks õige materjal ja tekstuur. Suits vajab keerukat materjali, et õigesti renderduda. Vali suur kuup ja mine materjalide kaardile. Seal määra materjali tüübiks *Volume* (mahuline) ja sea tiheduseks (*Density*) 0. Kui sead tiheduse väärtuse suuremaks kui 0, täitub toimimispiirkonna kuup üleni mahulise materjaliga. Samas mõjutavad suitsu ka [teised seaded](#). Me käsitleme neid hiljem.



Lisa tekstuur

Lisaks vajab suits ka oma teksturi. Blender 2.5-l on uus tekstuur just suitsu renderdamiseks, nimelt [voksliaandmed \(Voxel Data\)](#). Sa ei tohi unustada määrata toimimispiirkonna objekti ja muuta mõju.

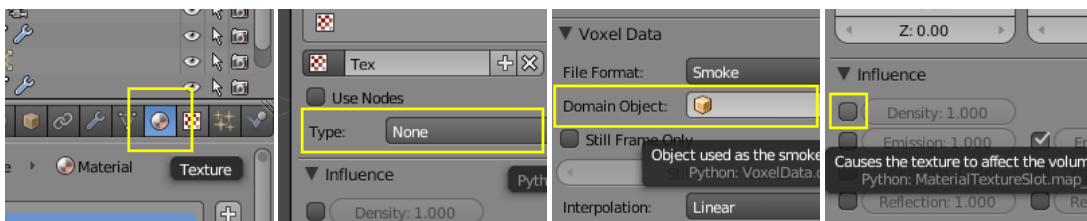


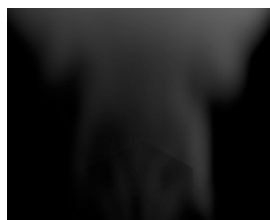
Mine teksturi kaardile ja määrä tüübiks *Voxel Data* (voksliandmed). Voxel Data seadete alt vali toimimispiirkonna (*Domain*) objektiks meie toimimispiirkonna kuup (selle nimi peaks olema lihtsalt *Cube*, kuna kasutame Blenderi vaikekuupi. Mõju (*Influence*) alt lülita sisse 'Tihedus' (*Density*) ja jäta selle väärtuseks 1000 (emissioon (*Emission*) peaks olema samuti automaatselt sisse lülitatud). Nüüd peaksid saama renderdada üksikuid kaadreid. Samuti saad valida suitsu värvi, lülitades "emissiooni värvi" (*Emission Color*) tagasi sisse.



Et näha suitsu selgemalt

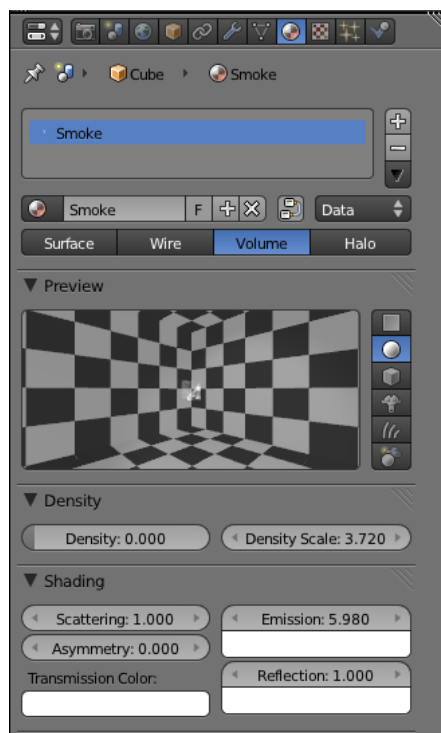
Vali maailma (*World*) kaardilt horisondi jaoks väga tume värv.



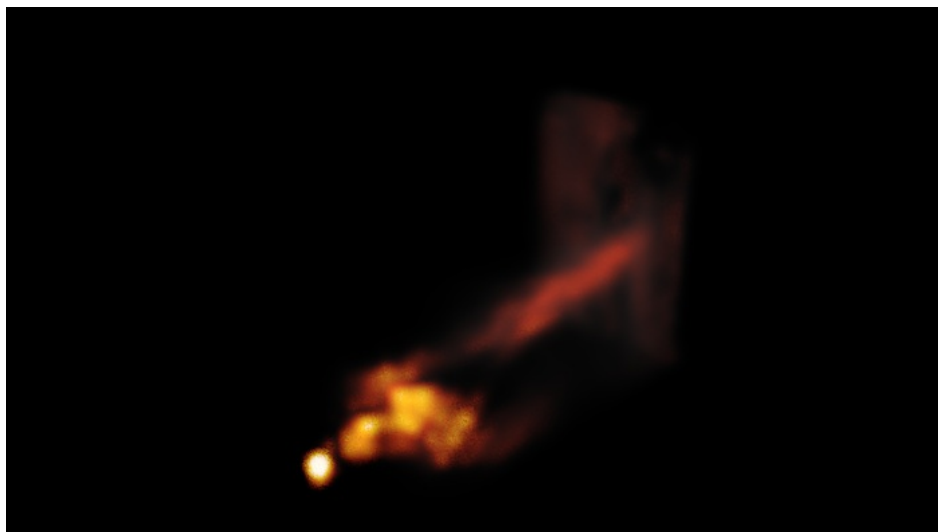
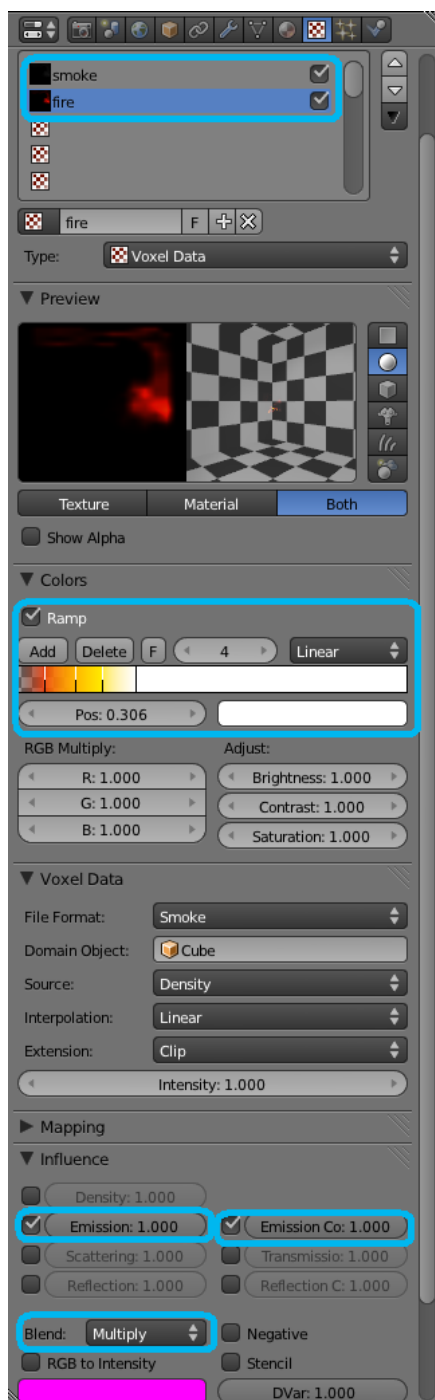


Suitsusimulaatori laiendamine: Tuli!

Saad muuta oma suitsu teise tekstuuri abil tuleks! Et teha tuld, keera materjalide paneelil emissiooni (*Emission*) väärtust juurde.



Siis lisa teine tekstuur. (Jäta vana tekstuur alles, muidu ei ole suitsu näha.) Anna sellele leekiv värviüleminek - katvusega (*alpha*) värvide põhjal ja muuda mõju emissiooniks (*Emission*) ja emissiooni värviks (*Emission Color*). Vali segamise (*Blend*) tüübiks Multiply (*korrutamine*).

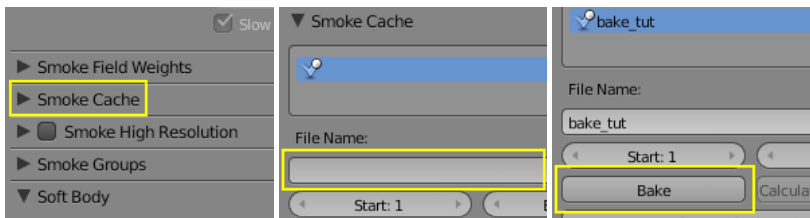


Suitsu simulatsioonide eeltöötlus

Kui soovid teha animatsiooni, pead oma suitsu esmalt eeltöötlemä. Eeltöötlus on lihtsalt simulatsiooni arvutamine. Et suitsu eeltöödelda, peab fail olema salvestatud. Arvutused salvestatakse vahemälu faili, millele saab ka nime anda. Mõnedel juhtudel võib Blender eeltöötlust tehes kokku joosta. [Vaata törkeotsingut]

Kerides mööda ajatelge või jooksutades animatsiooni 3D-vaates kiirklahviga AltA, oled sa juba teinud mälu reaalajas eeltöötlust. Aga et animatsiooni renderdada, peavad eeltöödeldud andmed olema kettal. Ja enne kui saad eeltöödelda, pead oma .blend-faili salvestama.

Järgmiseks vali toimetamispiirkonna kuup, mine füüsika (*Physics*) kaardile ja ava suitsu vahemälu sektsioon (*Smoke Cache*). Anna oma vahemälu failile nimi, sisestades see tekstikasti ja vajutades ↵ Enter. Vajutades nupule *Bake* (eeltöötle), arvutatakse simulatsiooni andmed ja salvestatakse kettale. Kas märkad, et ajatelje viga on kadunud? Nüüd peaksid sa saama animatsiooni renderdada.



Törkeotsing

K: Blender jookseb kokku, kui vajutan suitsu simulatsiooni nuppu *Domain* (toimimispiirkond)!

V: Selle põhjustavad vananenud draiverid. Uuenda oma draivereid.

K: Blender jookseb suitsu simulatsiooni eeltöödeldes kokku!

V: Sul saab selle arvutamisel operatiivmälu otsa. Proovi eeltöödelda madalama lahutusega.

K: Suits ei renderdu!

V: Mine tagasi ja loe dokumentatsiooni.

K: Kui proovin teha tuld, annab see imelikke tulemusi või ei paista üldse välja.

V: Veendu, et sul on materjalile määratud kõrge emissiooni (*Emission*) väärtus, et sul on suitsu tihedustekstuuri ja et oled tule tekstuuri segamisviisiks määranud *Multiply* (korruta).

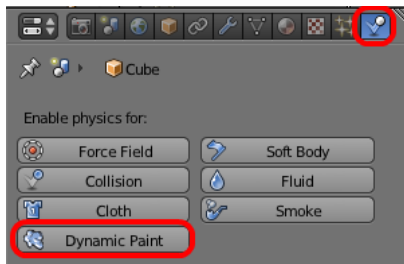
Välised lingid

- [Põhjalik sissejuhatus Blender 2.5 suitsu ja tule meetoditesse koos tavaliste vigade selgitusega](#)
- [MiikaH õpetus, kuidas teha Blenderis realistlikku tuld](#)

Dünaamiline maalimine (*Dynamic Paint*)

Dünaamiline maalimine (*Dynamic Paint*) on uus töötleja (*modifier*) ja füüsikasüsteem (*physics system*), mis võimaldab objekte muuta maalimise lõuenditeks ja pintsliteks ning luua tipuvärve (*vertex colors*), pildijadasid (*image sequences*) või nihkelaotust (*displacement*). See muudab võimalikuks mitmed efektid, mille saavutamine enne oli keeruline. Näiteks jalajäljed lumes, maad märjaks tegevad vihmapiisad, seinte külge jääv värv või objektid, mis ajapikku jäätuvad.

See juhend seletab lahti dünaamilise maalimise kasutajaliidese peamised omadused ja üldised võimalused.



Kuidas dünaamilist maalimist aktiveerida

Töötleja aktiveerimine

Dünaamilise maalimise saab lülitada sisse omaduste akna *Properties* füüsika sakist "Physics".

Tüübid

Töötlejal endal on kaks tüüpi:

[Lõuend \(Canvas\)](#)

Paneb objektid dünaamilise maalimise pintslitelt värvi vastu võtma.

[Pintsel \(Brush\)](#)

Paneb objekti lisama lõuendile värvi.

Märkus

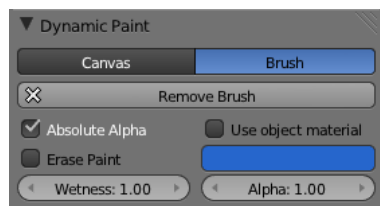
Võid pintslit ja lõuendi ka üheaegselt sisse lülitada. Sellisel juhul ei mõjuta sama objekti "pintsel" selle objekti "lõuendit", kuid võib siiski suhelda teiste stseenis olevate objektidega.

Vaata lisaks

- [Samm-sammuline tutvustus](#)
- [Detailne juhend, milles seletatakse piltide ja näidete abil lahti iga sätet](#) (Antud hetkel pole ajakohastatud)

Dünaamilise maalimise pintsell (*Dynamic Paint Brush*)

Peamine paneel



Pintsli peamine paneel

Esimesest pintsli paneelist saad määrata, kuidas pintsell lõuendi värvipindu mõjutab.

Absolute Alpha (absoluutne alfa)

See säte piirab pintsli läbipaistvuse (alfa) mõju. Ilma selleta "lisatakse" pintsli igas kaadris pinnale üha uuesti ja uuesti, suurendades seeläbi alfa ning seetõttu pintsli mõju lõuendile. Paljudel puhkudel aga soovitakse pintsli läbipaistvust mitte suurendada, kui see juba on pintsli määratud tasemel.

Erase Paint (kustuta värvi)

Paneb pintsli lisamise asemel olemasolevat värvi kustutama.

Wetness (märgus)

Määrab selle, kui "märg" uus värv on. Märgus on nähtav värvi pinnatüübi (*Paint*) märguskaardil (*Wetmap*). Tilkumise (*Drip*) ja valgumise (*Spread*) efektid sõltuvad samuti sellest, kui märg värv on.

Use object material (kasuta objekti materjali)

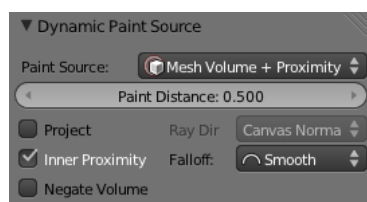
Sisselülitamisel saad määrata materjali, mida kasutatakse pintsli värvina. See tähistab nii materjali baasvärvi kui kõiki sellega lingitud tekstuure, mis lõpuks vastavad renderdatavale hajusvärvile (*diffuse color*) See säte on antud hetkel kasutatav ainult siis, kui kasutad Blenderi sisemist renderdajat *Blender Internal*.

Muul juhul saad allpool asuvast värvikastist määrata pintsli värvi.

Alpha (alfa)

Määrab pintsli alfa ehk nähtavuse. Alfa mõjutab ka lõpliku märgust.

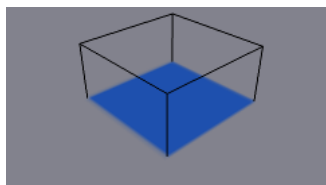
Allika paneel (*Source*)



Pintsli allika paneel

Pintsli allika sätted (*Source*) lasevad sul määrata, kuidas pintsli mõju/ristumist (*influence/intersection*) defineeritakse.

Praegu on valikus olemas viis pintsli käitumistüüpi, millest igaühel on täpsemaks määramiseks lisasätted:



Pintsli allikas (*Source*) - maht (*Volume*)

Võre maht (*Mesh Volume*)

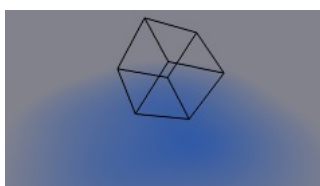
See on vaikesäte. Pintsell mõjutab kõiki võre mahus olevaid pinna punkte.

Lähedus (*Proximity*)

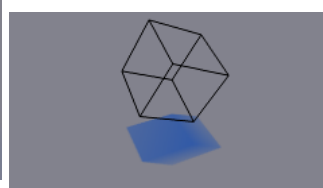
Kasutab ainult määratud vahemaad võre pinna lähima punktini. Pane tähele, et see ei pruugi mõjutada mahu sisemust, sest see ei ole pinnale lähedal. Läheduse hajumise (*Falloff*) tüüp võib olla sujuv (*Smooth*), terav (*Sharp*) või värvi ülemineku poolt määratud.

Project (projitseeri)

Projitseerib pintsli lõuendile määratud suunast. Põhimõtteliselt võib seda pidada "suuna järgi



Pintsli allikas (*Source*) - lähedus (*Proximity*). Pintsell mõjutab kõiki enda ümber olevaid lõuendi pikseleid



Sisselülitatud projitseerimise säte *Project*. Pintsell mõjutab lõuendit ainult normaali suunas

joondatud" läheduseks.

Võre maht + lähedus (*Mesh Volume + Proximity*)

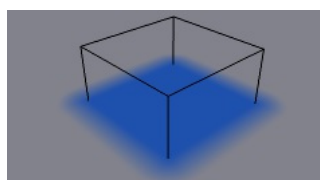
Sama nagu mahu tüüp, kuid see avaldab mõju üle kindlaks määratud kauguse. Kasutada saab samu hajumise tüüpe nagu läheduse (*Proximity*) puhul.

Inner Proximity (sisemine lähedus)

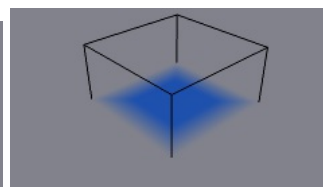
Lisab läheduse võre mahu sees.

Negate Volume (tühista maht)

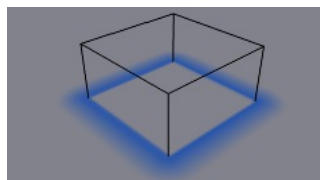
Tühistab võre mahu sees olles pintsli alfa.



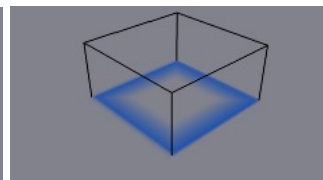
Mahu + läheduse (*Volume + Proximity*) pintsli ilma lisaäteteta



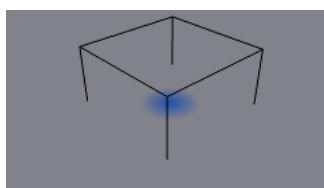
Sisemine lähedus (*Inner Proximity*). Läheduse hajumine on nüüd mahu sees nähtav



Tühista maht (*Negate Volume*).
Mahu sisemine külg on muutunud täiesti läbipaistvaks



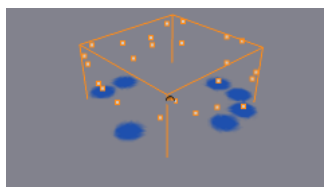
Sisemine lähedus (*Inner Proximity*) ja tühista maht (*Negate Volume*) korraga sisse lülitatuna



Pintsli allikas (*Source*) - objekti kese (*Object Center*)

Objekti kese (*Object Center*)

Selle asemel, et arvutada lähedus pintsli objekti võreni, mis võib mõningatel juhtudel olla suhteliselt aeglane, arvutatakse ainult kaugust keskpunktini. See on palju kiirem ja sageli piisavalt hea.

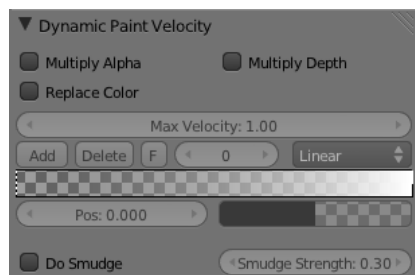


Pintsli allikas (*Source*) - osakeste süsteem (*Particle System*)

Osakeste süsteem (*Particle System*)

Pintsli mõju määravad valitud süsteemi osakesed.

Kiiruse paneel (*Velocity*)



Kiiruse paneel

Selles paneelis on ära toodud pintsli sätted, mis sõltuvad objekti kiirusest.

Üleval otsas on värvilemineku ja mitu sellega seotud sätet. Värvilemineku kujutab põhimõtteliselt pintsli kiiruse väärtuseid: vasak pool on nullkiirus ja parem pool on maksimumkiirus (*Max velocity*). Kiirust mõõdetakse "Blenderi ühikutes kaadri kohta".

Üleval olevaid linnukese kaste saab kasutada ülemineku mõju määramiseks.

Multiply Alpha (korruta alfa)

Kasutab hetke kiirusest sõltuvalt värvi ülemineku alfa väärtust ja korrutab selle pintsli alfaga.

Replace Color (asenda värv)

Asendab pintsliga värvi ülemineku värviga.

Multiply Depth (korruta sügavust)

Korrutab pintsliga sügavuse ristumise (*depth intersection*) efekti. Põhimõtteliselt saad muuta nihke (*displace*) ja laine (*wave*) tugevust sõltuvalt pintsliga kiirusest.

Määride (*Smudge*) sätted

Määride (*Smudge*) paneel pintsliga liikudes pinnal olemasolevaid värve plekiliseks muutma või määrima. Selle sätte tugevust saab määrata valiku *Smudge Strength* alt.

Isegi kui määrimine on sisse lülitatud, teeb pintsel ikkagi oma tavapärasest värvimise efekti. Kui soovid ainult määrivat pintslit, kasuta null alfati. Samuti võib koos määrimisega olla sisse lülitatud kustutamise säte (*Erase*).

Lainete paneel (*Waves*)

Pintsliga lainete paneel

Seda paneeli kasutatakse pintsliga mõju määramiseks pindade "lainetamisel".

Saad kasutada laine tüübi menüüd *Wave Type* määramaks seda, millist mõju pintsel lainete simuleerimisele avaldab. Allpool on kaks sätet edasiseks muutmiseks.

Factor (faktor)

Määrab, kui tugevasti pintsliga sügavus (*depth*) mõjutab simulatsiooni. Saad kasutada ka negatiivseid väärtuseid, et pintslit vee surumise asemel seda üles tõmbama panna.

Clamp Waves (piira laineid)

Mõnedel juhtudel läheb pintsel väga sügavale pinna sisse, keerates terve simulatsiooni vussi. Saad kasutada seda sätet, piiramaks mõju teatud sügavusega.

Laine tüüpe (*Wave Type*) on neli tükki:

Depth Change (sügavuse muutus)

See valik paneb pintsliga laineid tekitama, kui pinnaga ristumise sügavus selles punktis "muutub". Kui pintsel on liikumatu, ei ole sellel mõju.

Negatiivse faktori (*Factor*) kasutamisega selle tüübi puhul saab tekitada väga kena "kiiluvee" efekti liikuvatele objektidele, nt laevadele.

Obstacle (takistus)

Mõjutab ristumisel pidevalt pinnast. Selle pintsliga tüübi puhul peegelduvad lained ka pintslilt eemale. Kuid tulenevalt laine simulatsiooni algoritmist tekitab see tüüp laine pinnale ebaloosuliku "muhu", kui pintsel paigal seisab.

Force (jõud)

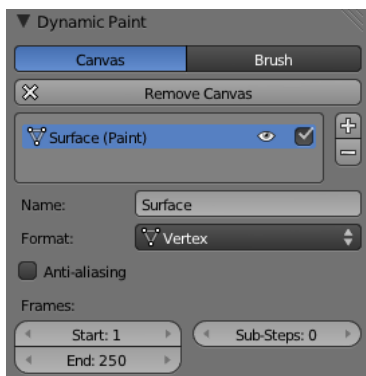
Mõjutab otseselt laine liikumise kiirust. Seetõttu ei vasta see efekt üks-ühele pintsliga ristumissügavusele, kuigi jõu tugevus sellest sõltub.

Reflect Only (ainult peegelda)

Sellel tüübil pole pelgalt pinnale mingit nähtavat mõju, vaid see peegeldab laineid, mis on juba pinnal.

Dünaamilise maalimise lõuend (*Dynamic Paint Canvas*)

Peamine paneel



Lõuendi peamine paneel

Lõuendi esimene paneel sisaldab dünaamilise maalimise pindade (*Dynamic Paint surfaces*) nimekirja. Need pinnad on põhimõtteliselt värvi kihid, mis töötavad üksteisest eraldi. Sa saad määrata nende sätteid ja neid eelrenderdada (*bake*) eraldi.

Kui pinna tüüp/formaat võimaldab tulemuse eelvaatamist 3D-vaatekanas, on nähtav silma ikoon, mille abil saab eelvaatamist sisse ja välja lülitada.

Linnukese kast määrab, kas pind on aktiivne või mitte. Kui see pole valitud, siis ei tehta arvutusi või eelvaateid üldse.

Igale pinnale saad ka lihtsaks identifitseerimiseks oma nime anda.

Altpoolt saad määrata pinna tüübi ning muuta kvaliteedi (*quality*) ja ajastuse (*timing*) sätteid.

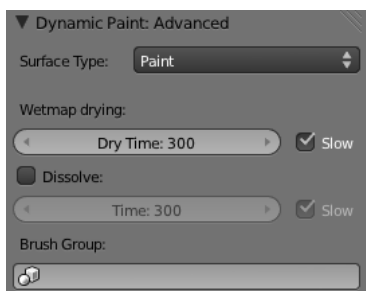
Igal pinnal on kindel formaat ja tüüp. Formaati (*format*) määrab, kuidas andmeid salvestatakse ja väljastatakse. Praegusel hetkel on olemas kaks formaati:

- Pildijadad (*Image Sequences*). Dünaamiline maalimine genereerib väljundina kindlaksmääratud resolutsiooniga UV-laotatud pildifaile.
- Tipp (*Vertex*). Dünaamiline maalimine muudab otseselt võre tippude andmeid. Tulemused salvestatakse punkti puhvrina (*point cache*) ja neid saab vaateakendes kuvada. Aga tippude tasemel töötamine nõuab töötamiseks ka korralikult tükeldatud võret.

Kvaliteedi sätete alt saad muuta pildi resolutsiooni (pildijadade puhul) ning sakisilumist (*anti-aliasing*).

Seejärel saad määrata pinna arvutamise algus- ja lõppkaadri ning kasutatavate alamsammude (*sub-steps*) arvu. Alamsammud on kaadrite vahel võetavad sãmplid ehk proovid, mida läheb tavaliselt vaja siis, kui pintsli on väga kiire.

Lisasätete paneel (*Advanced*)



Lõuendi lisasätete paneel

Lisasätete paneeli *Advanced* kaudu saad muuta pinna tüüpi ja sellega seotud sätteid.

Igal pinnal on "tüüp", mis määrab selle, milleks pinda kasutatakse. Kasutatavad tüübid on:

- Värv (*Paint*)
- Nihe (*Displace*)
- Lained (*Waves*)
- Kaal (*Weight*)

Ühised sãtted

Igal pinna tüübil on eraldi sãtted, mida muuta. Enamus tüüpidel on olemas ka hajumise (*Dissolve*) ja pintsli (*Brush*) sãtted:

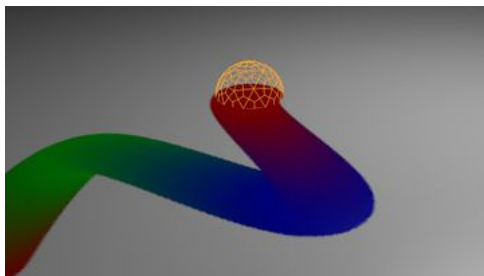
Dissolve (hajumine)

Seda kasutatakse määratud aja jooksul pinna sujuvaks originaalolekusse tagasi hajutamiseks.

Brush Group (pintsli grupp)

Kasutatakse kindla objektigrupi määramiseks, millest võetakse pintsli objektid

Värv (*Paint*)

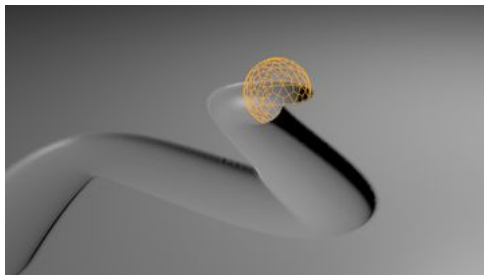


Värvi pind

Värvi pind *Paint* on kõige tavalisem pinna tüüp, mille väljunditeks on värvi (*color*) ja märguse (*wetness*) väärtused. Tipu pindade (*vertex surfaces*) puhul on tulemuseks tipuvärvid (*vertex colors*) väljund.

Märguskaart (*Wetmap*) on mustvalge väljund, mis kujutab värvi märgust. Valge tähistab maksimaalselt märga, must täiesti kuiva. Seda kasutatakse tavaliselt renderdamisel maskina. Mõned "värviefektid" mõjutavad ainult märga värvi.

Nihe (*Displace*)



Nihke pind

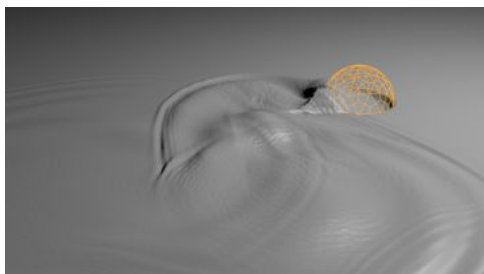
See pinna tüüp väljastab pintsliobjektide ristumissügavuse (*intersection depth*).



Vihje

Kui nihke väljund näib liiga kare, aitab tavaliselt silumistööriista *Smooth* lisamisest dünaamilise maalimise (*Dynamic Paint*) järele töötajate pinusse.

Lained (*Waves*)



Lainete pind

See pinnatüüp tekitab simuleeritud lainete liikumist. Nagu nihkegi, kasutab ka lainete pind pintslit ristumissügavust (*intersection depth*), et määrata pintslitugevust.

Liikumise muutmiseks saad kasutada järgnevat sätteid:

Open Borders (avatud piirid)

Lubab lainetel ületada võre "servi", selle asemel et neilt tagasi peegelduda.

Timescale (ajaskaala)

Muuda otseselt simulatsiooni kiirust, mõjutamata simulatsiooni tulemust. Madalamad väärtused muudavad simulatsiooni aeglasemaks ja vastupidi.

Speed (kiirus)

Määrab, kui kiiresti lained pinnal liiguvad. See säte vastab ka simulatsiooni suurusele. Poole väiksem kiirus on võrdne kaks korda suurema pinnaga.

Damping (sumbumine)

Vähendab aja jooksul lainete tugevust. Põhimõtteliselt määrab selle, kui kiiresti lained kaovad.

Spring (vedru)

Määrab jõu, mis tõmbab vee tagasi nulltasemele (*zero level*).

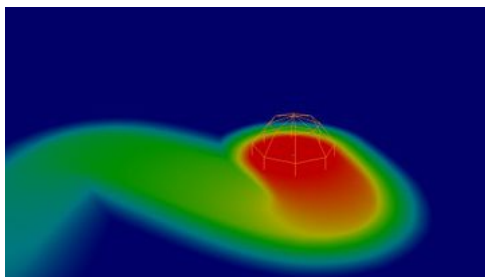


Vihje

Mõnedel puhkudel muutub laine liikumine pintslit ümbruses väga ebastabiilseks. Selle vastu aitab tavaliselt laine kiiruse, pintslit

laine faktori (*wave factor*) või isegi võre/pinna resolutsiooni vähendamine.

Kaal (*Weight*)



Kaalu pind

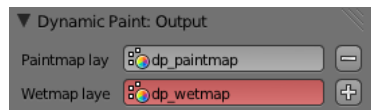
See on eriline pinna tüüp, mida saab kasutada ainult koos tipu formaadiga. See väljastab tippude kaalugruppe, mida saab kasutada koos teiste Blenderi töötlejate ja tööriistadega.



Vihje

Kaaluga pindade puhul on tavaliselt parem kasutada lähedusel (*proximity*) põhinevaid pintsleid, et kaalu väärtuste vahel toimuks sujuv vähenemine.

Väljundi paneel (*Output*)



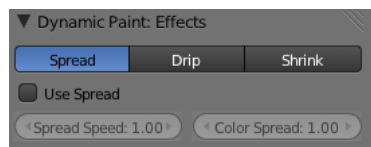
Lõuendi väljundi paneel

Väljundi paneelist *Output* saad sa määrata, kuidas pind oma tulemused väljastab.

Tipu formaadiga pindade puhul saad valida võre andmekihi (värv / kaal sõltuvalt pinna tüübist), millele tulemused genereeritakse. Saad kasutada "+" / "-" ikoone, et valitud nimega andmekihte lisada/kustutada. Kui valitud nimega kihti ei eksisteeri, kuvatakse seda punasena.

Pildijada formaadiga pindade puhul saad määrata UV-kihi ning väljundfaili salvestamiskataloogi, faili nime ja pildi formaadi.

Efektide paneel (*Effects*)



Lõuendi efektide paneel

See on värvi tüüpi pindade (*Paint*) eripaneel. See loob lõuendi pinnal animeeritud liikumisi.

Praegu on olemas kolm efekti:

Spread (voola)

Värv voolab aeglaselt ümbritsevate punktideni, täites lõpuks kõik ühendatud alad.

Drip (tilgu)

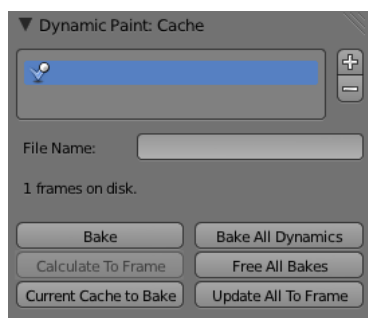
Värv liigub kindlas suunas, mille määravad Blenderi jõuväljad (*force fields*), gravitatsioon ja kiirus (*velocity*), mille mõju määrab omakorda kasutaja.

Shrink (kahane)

Värvitud ala kahaneb aeglaselt, kuni kaob täielikult.

Voolamise ja tilkumise efektide puhul mõjutatakse ainult "märga värvi", nii et mida enam värv kuivab, seda aeglasemaks liikumine muutub, kuni lõppeb täielikult.

Puhvri paneel (*Cache*)



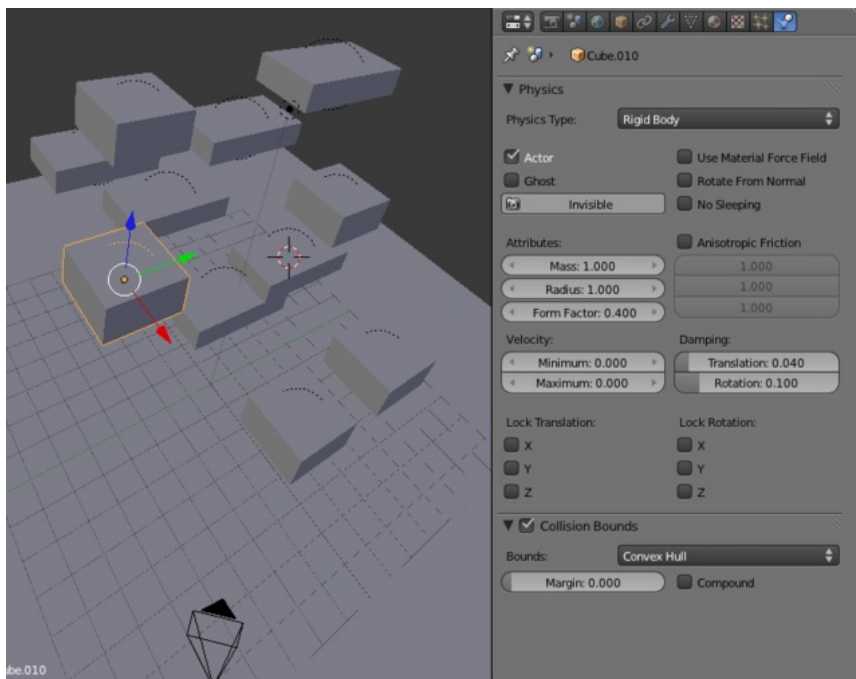
Lõuendi puhvri paneel

See on praegu nähtav ainult tipu formaadiga pindade puhul. Selle abil saad punktide puhvrit (*point cache*) muuta ja eelrenderdada (*bake*).

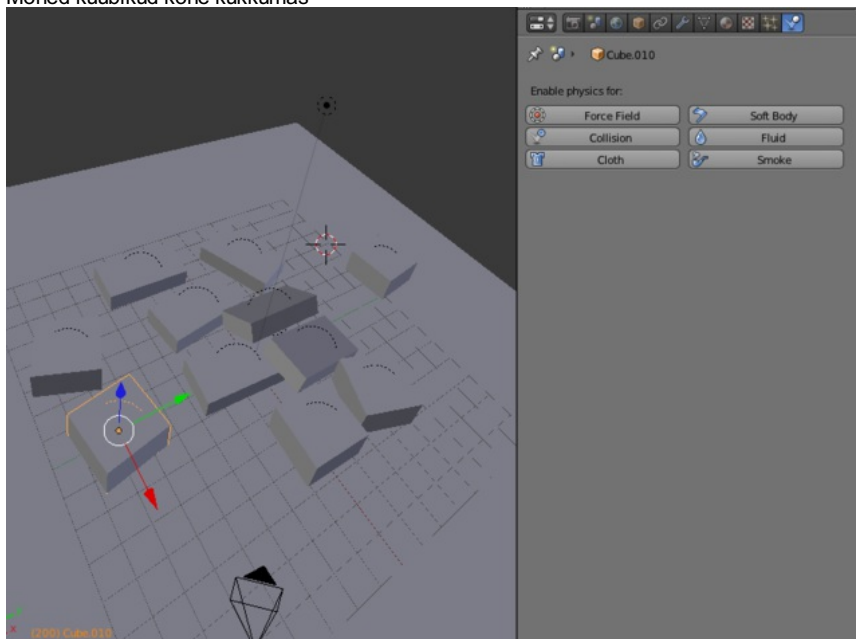
Mängumootori füüsika teisendamine

Vahel võid sa tahta animeerida müüri purunemist kokkupõrkel objektiga või siis hulka objekte kokku varisemas, kukkumas või põrkamas reaalse füüsika reeglite järgi. Selleks võid sa sisestada juhtkaadrid manuaalselt ja seadistada füüsika ning kiirenduse simuleerimiseks käsitsi katse-eksituse teel F-kõveraid. Või siis teha seda oluliselt lihtsamalt ja automaatselt, kasutades ära Blenderi mängumootori füüsikat. Blenderil on nüüd võimalus salvestada mängumootori animatsioone ja teha nende põhjal animatsiooni juhtkaadrid.

Animatsiooni saab salvestada, minnes menüüsse *Game>Record Animation* (mäng->salvesta animatsioon). Animatsiooni saab siis salvestada kiirklahvidega Alt+A



Mõned kuubikud kohe kukkumas



Hunnik

Kui sul on vaja lihtsalt staatilist hunnikut, siis võid sa minna viimasesse kaadrisse ja kustutada kiirelt kõik juhtkaadrid, muutes nad NLA-deks ja kustutades.

Simulatsiooni jõudluse parandamine

OpenMP (Mac OSX)

Kuidas kasutada jagatud appliskripti (*distributed applescript*), et OSXi operatsioonisüsteemis simulatsiooni jõudlust parandada

Vähene eelrenderduse jõudlus

Sageli kohtad sa avatud hulgitöötlust (*openMP* ehk OMP) kasutades hargtöötlusega (*multithreaded*) simulatsioonide (eriti vedelike puhul) vähest eelrenderduse (*baking*) jõudlust.

See tuleneb sellest, kuidas domeeni jagamine (*domain splitting*) saadab harud (*threads*) rakkudesse (*cells*), mis pole mõnikord "täidetud", mille tulemusena ei anna suurem harude arv mitte mingisugust võitu kiiruses. See sõltub enam-vähem simulatsiooni resolutsioonist ja objektide tihedusest.

Kui sul on selline "ebakasulik" simulatsioon, siis aita kõvasti see, kui vähendad OMP harude arvu väikese väärtuseni, mitte ei lase OMP-l lihtsalt kasutada kõiki saadaolevaid tuumasid.


Lahendus

OSXi openMP toega Blenderis saad nüüd kasutada appliskripti, mis määrab OMP-harude kasutuse. See muudab terminali kasutamise mittevajalikuks.

Vaikeväärtus on praegusel hetkel neli tuuma.

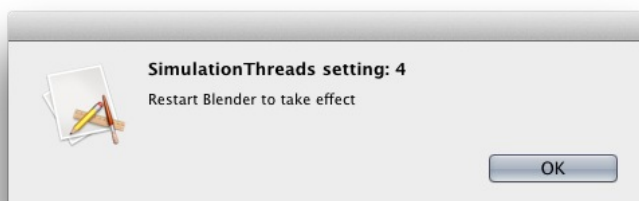
Kui sa jätad sisendi välja tühjaks, siis leiab see skript sinu loogiliste protsessorituumade arvu (ka HyperThreading). Sama teeks ka openMP, kui tal ei oleks määratletud keskkonna muutujaid.

Appliskripti kasutamise sammud

1. Ava oma Blenderi kaustas "set_simulation_threads" nimeline appliskript  `set_simulation_threads`
2. Määra soovitud harude arv (jätta tühjaks, kui soovid, et OMP saaks kasutada kõiki lubatud tuumasid)



3. Vajuta uue väärtuse kinnitamiseks o.k., mille järel näidatakse sulle teadet uue sätte aktsepteerimise kohta.



4. Sule Blender ja ava uuesti, et säte tööle hakkaks.
5. Kontrolli eelrenderduse kulgu ja simulatsiooni kaadrisagedust ning vajaduse korral korda samme 1–4, et leida optimaalne väärtus.

Renderdamine

Renderdamine on ruumilise graafika viimane protsess (kui järeltöötlus kõrvale jätta) ja faas, milles luuakse sinu ruumilisele stseenile vastav kahemõõtmeline pilt. Renderdamine on väga protsessorinõudlik tegevus. Renderdamiseks võib kasutada ainult üht arvutit või paljusid arvuteid korraga. Mitme arvuti vahel saab ära jagada ühe renderdatava pildi erinevad segmendid. Animatsiooni puhul saab arvutite vahel ära jagada terved kaadrid. See peatükk annab täieliku kirjelduse nendest Blenderi omadustest, mis on seotud pildi või animatsiooni loomisega.

Pärast materjalide, tekstuuride, valgustite ja kaamera paikapanekut võib alustada renderdamist. Suure tõenäosusega ei ole esimene renderdatud pilt selline nagu tarvis: ole valmis paljudeks testrenderdusteks. Siin peatükis kirjeldame valikuid ja seadeid, mida renderdusel soovitud pildikvaliteedi saavutamiseks kasutada saame.

Blender kasutab sisseehitatud renderdusmootorit. See on kiire renderdaja ja annab oskusliku viimistlemisega häid tulemusi. Saadaval on ka teisi väliseid renderdajaid, mis pakuvad alternatiivseid renderdustööriistu.

Teame, et üle maailma kasutatakse meie programmi väga erineva jõudlusega arvutitel. Renderdamine on see üks arvutigraafika loomise protsess, mille jaoks pole kunagi piisavalt protsessorijõudlust ega kettaruumi. Korporatiivkeskkonnas on väga lihtne täita terabaidne server, kui laadida sinna sinna kümme tundi DV-lindilt pärit videoid ja neid veidi monteerida. Suure töö saab nõrgal arvutil pildikvaliteeti ohverdamata valmis renderdada seda osadeks jagades.

Selgitame renderdamise paneelil olevat ja täpsustame seda edasistes peatükkides.

Ülevaade

Valitud stseeni renderdamiseks vajuta Image-nuppu Render-paneelil. Seda saab teha ka, vajutades klaviatuurilt F12 (renderdatud pildi ekraanil kuvamise viise saad valida [renderdamise väljundi seadete alt](#)). Vaata ka [renderdamise akent](#).

Video saad, vajutades nupule Animation. Tulemust hoitakse mäluhuphris ja kuvatakse eraldi aknas. Salvestada saab, vajutades klaviatuuril F3 klahvi või valides menüüst File->Save Image, kasutades Output-paneelil määratud seadistusi. Animatsioone salvestatakse vastavalt kasutaja valikutele, tavaliselt kaadrite seeriana kasutaja määratud kausta. Vaata peatükke [Väljundi valikud](#) ja [Animatsiooni renderdamine](#).

Pilt renderdatakse sellise suurusega, nagu on määratud paneelil Dimensions.

Töövoog

Renderdamise protsess on üldiselt järgmine:

1. Kõikide stseeni objektide loomine
2. [Stseeni valgustamine](#)
3. [Kaamera paikapane](#)
4. Ilma sakisilumiseta ja kiirtejälituseta proovirenderdus 25% juures, et vältida suurt ajakulu
5. Materjalide, tekstuuride ja valgustuse täiustamine
6. Eelneva kordus, kuni tulemus on hea
7. Järjest suurema resolutsiooniga renderdamine ja igal korral väikeste täienduste tegemine, kuni lõpuks on käes täisresolutsiooniga hea pilt
8. Pildi salvestamine

Renderdusfarm

Renderdusprotsessile suurema koguse protsessoriaja saamiseks on mitmed võimalusi.

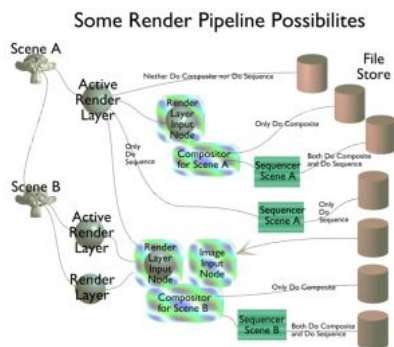
Esiteks tuleb mitmetuumalise protsessori korral määrata, et Blender kasutaks kõiki tuumasid.

Teiseks võib kohtvõrgus olevate arvutite vahel renderdustöö kaadrite kaupa ära jagada. Näiteks kui on olemas 5 sarnase jõudlusega arvutit, võib teha nii, et esimene renderdab kaadrid 1–40, teine 41–80 jne. Kui üks arvuti on aeglasem, võib sellele lihtsalt vähem kaadreid jätta. Kohtvõrgus renderdamiseks muuda .blend-faili sisaldav kaust jagatud kettaks. Pane sinna ka kõik .blend-faili välised failid nagu tekstuurid jm. Käivita iga arvuti peal Blender ja ava vastav .blend-fail. Muuda igas arvutis renderdamise algus- ja lõpukaader ära, aga ära faili salvesta. Käivita renderdamine. Kui kasutasid väljundis suhtelist väljundi rada kirjeldavat sümbolit, paigutatakse kõikide arvutite poolt renderdatud kaadrid samasse kausta.

On võimalik kasutada ka sellist renderdamist, et pakid oma .blend-faili koos muude vajalike failidega kokku ja saadad sõpradele e-postiga, failijagamislehekülje või Verse'i kaudu internetti renderdamiseks laiali. Nemad renderdavad oma osa ja saadavad saadud kaadrid tagasi. Abivalmis sõprade puhul saab nii koostööd teha küll.

Neljandaks võib renderdada renderfarmi teenusepakkuja juures. Näiteks toome teenuspakkuja BURP või tasuta Renderfarm.fi, mis on ühingute poolt tööle pandud. Neile võib saata e-kirja koos oma failiga ja nemad siis jaotavad renderdustöö oma arvutite vahel ära. BURP ja Renderfarm.fi on siin ära märgitud selle pärast, et see on tasuta ja see kasutab vennaslike Blenderi kasutajate arvutiteid BOINC-tüüpi taustaprotsessina. Teised teenusepakkujad võtavad raha mingi perioodi eest või täpselt kasutatud aja eest.

Renderdustööde ühendamine



Blenderil on renderdamiseks 3 erinevat osa, mida saab kasutada üksikult või ühendada "tootmisliiniks".

- Renderdusmootor
- [Komposiitor](#)
- [Järjestaja](#)

Neid võib eraldi kasutada, aga neid võib ka üksteisega seotuna (lingituna) kasutada. Näiteks võib kasutada ainult järjestajat ennast mingi video töötlemiseks. Võib kasutada ka ainult komposiitorit, et näiteks mingi pildi toone muuta. Stseeni võib renderdada ja salvestada otse ilma komposiitorit või järjestajat kasutamata. Need võimalikud variandid on kuvatud pildi ülaoas paremal.

Samuti võib stseenid kokku linkida kas otse või vaheetapina salvestusmeediat kasutades. Igal stseenil võib olla mitu renderduskihti ja igat renderduskihti saab komposiitoris kokku miksida. Aktiivne renderduskiht on see, mida parajasti kuvatakse ja mis on aktiivseks märgitud. Kui parajasti kuvatav renderduskiht ei ole märgitud/aktiveeritud, kasutatakse renderdamiseks nimekirjas järgmist märgitud renderduskihti. Kui nupud Compositing ja Sequencer on väljas, kuvatakse pilt lõpliku renderdusena.

Kui Compositing on sisse lülitatud, suunatakse renderduskihid komposiitorisse. Need käivad siis läbi määratud nuudlite ja jõuavad komposiitori väljundisse, kust saab selle salvestada või saata edasi järjestajasse, kui *Do Sequence* on sisse lülitatud.

Kui järjestaja (Sequencer) on sisse lülitatud, siis suunatakse komposiitori väljund (juhul kui *Do Composite* on sisse lülitatud) või aktiivne renderduskiht (juhul kui *Do Composite* ei ole sisse lülitatud) järjestaja stseeniribaks. Seal töödeldakse seda vastavalt järjestaja seadistustele ja lõpuks saab sellest stseeni kaadripilt.

Juhul kui .blend-failis on mitu stseeni, näiteks stseen A ja stseen B, on asi mõnevõrra keerulisem. Kui stseenis B on Compositing sisse lülitatud, siis saab renderduskihi sõlme suunata stseeni A renderduskihiks. Tuleb ära märkida, et siis ei ole see pilt järeltöödeldud. Kui soovid näiteks stseeni A väljundit saada stseeni B järeltöödelduna, siis tuleb kõigepealt stseen A kettale renderdada ja siis stseeni B komposiitori pildisisendi sõlme abil sisse tuua.

Alumine osa joonisest näitab Blenderi maksimaalset kasutamist: järeltöödeldud pildid ja dünaamiline renderduskiht stseenist A on komposiitoris miksitud kahe renderduskihiga stseenist B. Seejärel toimub järjestajas montaaž ja lõpuks salvestatakse tulemus.

Need näited on ainult väike osa Blenderi kasutamise võimalustest. Jätka selle teema lugemist ja hiljem harjutades saad luua oma ainulaadse töövoogu.

Renderduse seadistuste paneel

Renderdamise sakk kuvab kõik sisseehitatud renderdaja valikud. Valides renderdamiseks välise renderdaja, kuvatakse seal vastavalt välise renderdaja seadistused.

Render

Selle nupu abil saab käivitada nii [üksiku pildi renderdamise](#) kui ka [animatsiooni renderdamise](#).

Siit saab ka valida renderdatava pildi salvestamise kohta. Neid kirjeldatakse lehel [Renderdusvaade](#).

Kihid

Renderduskihtide kohta leiad täpsemat infot lehelt [Renderduskihid](#) ja [Renderduskäigud](#).

Mõõtmed

Selles menüüs on pildi suuruse seadistused (vaata [Väljundi seaded](#)) ja valikud järjestajas renderdamiseks (vaata [Animatsioonid](#)).

Sakisilumine (*Anti-Aliasing*)

[Sakisilumisega](#) renderdamine on kvaliteetseks tulemuseks väga tähtis. Et objektide servad ja jooned ei oleks "treppis" ega "karvased".

Liikumishägu (*Motion Blur*)

[Liikumishägu](#) efekt on animatsioonide juures väga tähtis. Selle abil saab animatsioonid usutavamaks muuta. Ilma selleta oleksid ka liikuvad objektid liiga selged ja ebarealistlikud.

Varjutamine (*Shading*)

Nende seadistustega saab määrata, milliseid varjutamise aspekte renderdamisel arvestatakse. Mittemärgituks muutmine lülitab need välja.

- [Tekstuurid](#)

- [Varjud](#)
- [Pinna sisehajuvus](#)
- [Keskkonnavastendus](#)
- [Kiirtejälitus](#)
- [Värvihaldus](#)

Kasutab sisselülitamisel lineaarset töövoogu

- [Alfa](#)

Määrab, kuidas läbipaistvad pikslid renderdada

Väljund

Määrab, kuhu renderdatud pildid salvestatakse ja millist failitüüpi kasutatakse. Vaata [Väljundi valikud](#).

Jõudlus

Määrab arvuti protsessori ja mälu kasutuse. Vaata [Jõudluse peatükki](#).

Järeltöötlus (*Post Processing*)

Sellega saab juhtida pärast renderdamist pildile lisatavaid efekte. [Komposiitori](#) või [järjestaja](#) kasutamisel saad määrata, et Blender kasutaks sealseid efekte.

Video jaoks saab renderdada poolkaadrite kaupa [Video tarbeks renderdamine](#)

[Virtoonimine \(*Dithering*\)](#) on pikslite hägustamise meetod.

Visandlike või multifilmilike äärejoonte jaoks võib sisse lülitada [servajoone renderdamise](#).

Tempel (*Stamp*)

[Tembeldamine](#) lisab renderdatud pildi peale infotekste.

Eelrenderdamine (*Bake*)

[Eelrenderdamine](#) on selline protseduur, kus näiteks vari, valgustus või muud tüüpi varjutus pannakse n-ö tekstuuri enda sisse. See on sobilik reaajas (näiteks arvutimängud) toimuva 3D-graafika jaoks, sest siis ei pea graafikaprotsessor nii palju arvutama kui muidu.

Kaamera

Kaamera (Camera) on objekt, mille abil Blender renderdab. Kaamerad määravad, milliseid stseeni osi renderdamisel näidatakse. Stseenis võib olla mitu kaamerat, kuid renderdamiseks on vaja vähemasti üht.

Uue kaamera lisamine

Mode: Objektirežiim (Object mode)

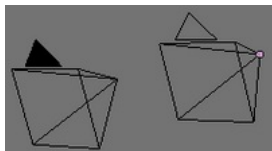
Menu: Add » Camera

Vajuta objektirežiimis ⇧ ShiftA ja vali Add » Camera.

Aktiivse kaamera vahetamine

Mode: Objektirežiim (Object mode)

Hotkey: Ctrl0 NumPad



Aktiivne kaamera (vasakul).

"Aktiivne" kaamera on see kaamera, mille abil renderdatakse ja milles paikneb kaameravaade (0 NumPad). Kaamera aktiveerimiseks vali soovitud kaamera, vajuta Ctrl0 NumPad (samaaegselt muudetakse ka 3D-vaade kaameravaateks). Kaamerata stseeni ei saa renderdada.

Aktiivse kaamera eristamiseks on 3D-vaates ülasuunda näitav kolmnurk seest täidetud. Vasakpoolne kaamera pildil *Aktiivne kaamera (vasakul)*.



Aktiivne kaamera, nagu ka kihid, võivad olla seotud konkreetse vaatega või olla üldised: (lukustatud) avatud stseeni jaoks – vaata [stseeni lukustamise osa 3D-vaate valikute lehelt](#).

Kaamera seaded

Mode: Objektirežiim (Object mode)

Panel: Camera



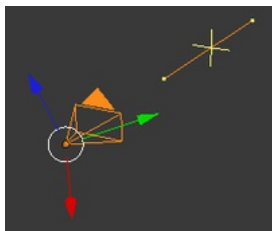
Paneel Camera.

Kaamerad on renderdamisel nähtamatud: seetõttu puudub neil materjal ning tekstuur. Kaameral on samas Object- ja Editing-seadistuste paneel, mida kuvatakse siis, kui kaamera on valitud.

Lääts (Lens)

- Perspective/Orthographic (perspektiiv/paralleel)

Lülitab paralleelprojektsiooni (Orthographic) ja perspektiivprojektsiooni (Perspective) vahel. Paralleel- ja perspektiivprojektsiooni kohta vaata täpsemalt peatükkidest [3D-vaade](#) ja [Perspektiiv](#).



Kaamera sisselülitatud nähtava ala ja fookuspunktiga.

- Focal Length (fookuskaugus)

Fookuskaugus millimeetrites või kraadides. Pärast paralleelprojektsiooni valimist muutub fookuskauguse valik

paralleelprojektsiooni suurendusastmeks (Orthographic Scale). Sellega määratakse kaamera nähtava ala suurus.

- Panorama (panoraam)

Renderdab stseeni panoraamvaate jaoks, silindrilise kaameraga.

- Shift X/Y (nihe X/Y)

Nihutab kaamera nägemisala. Selle abil nihutamist tuleks enamasti vältida, sest Shift abil ei saa muuta kaamera enda asukohta, vaid ainult nähtavat pilti kaamera suhtes nihutada. Kasutatakse stereopiltide loomisel.

- Clipping Start/End (nähtava ala algus ja lõpp)

Nägemisulatuse alguse ja lõpu määramine Renderdatakse ainult määratud vahemikku jäävad objektid. Kui valik Limits on sisse lülitatud, kuvatakse kaamera nägemisteljel kaht joonega ühendatud kollast täppi. Need näitavad renderdatavat vahemikku.

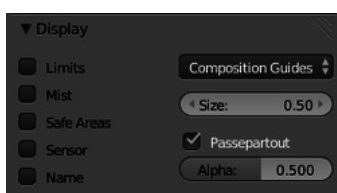
- Depth of Field (sügavusteravuse)

[Sügavusteravuse](#) kasutamisel määrab fookuspunkti määratud objekt. Siin määratu deaktiveerib kaugusnupul valitu.

- Distance (kaugus)

Kaugus fookuspunktini. Seda näidatakse kollase ristikesena kaamera nägemisteljel. Valik Limits peab selle nägemiseks olema sees. Kasutamiseks kombineerituna [Komposiitorisõlmega Defocus](#).

Esitus (*Display*)



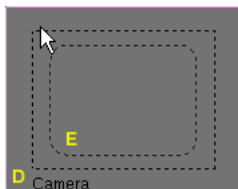
Kaamera esitluse paneel.

- Limits (piirid)

Lülitab piirangute kuvamist sisse-välja.

- Mist (udu)

Lülitab sisse-välja udu piiride näitamise. Piirangud muutuvad nähtavaks kahe valge punktina kaamera vaateteljel (**A** ja **B** kaamerat kujutaval pildil). Udu piiramise parameetrid leiab kontekstist World, paneelilt Mist.



Kaameravaade, kus on kuvatud kaamera nimi ja titreerimiseks ohutu ala.

- Title Safe (tiitrivaba ala)

Kui see on sisse lülitatud, näidatakse kaameravaates veel üht katkendjoonega raami. Tähtsad objektid, nagu näiteks tiitrid, tasub teles jätta selle ala sisse.

- Name (nimi)

Lülitab kaamera nime kuvamist sisse-välja.

- Size (suurus)

Kaamera võrestiku suurus 3D-vaates. Selle muutmise ei mõjuta renderdamistulemust ja on kaamera teiste objektide suhtes "parajaks" timmimiseks. Kaamera suurus saab muuta ka tavalisel moel S-klahvi abil mõõtkava muutes.

- Passepout, Alpha (aspartuu, alfa)

Nende valikute abil saab kaamera nähtavast alast väljaspool oleva raami tumedamaks teha, vastavalt Alpha väärtusele.

Märkus

3D-vaate (3D View) akna jaoks on olemas samasugused valikud nagu kaameragi jaoks. Nendeks on Orthographic/Perspective ja Clip Start/Clip End. Need seadistused ei mõjuta kaamera poolt renderdatavat pilti, vaid ainult 3D-vaate aknas olevat vaadet. Need seadistused saab kätte 3D-vaate vaatemenüüst View. Vaata lähemalt [3D-vaate valikute peatükist](#).

Kompositsioonijooned

Kompositsioonijooned (Composition Guides) on saadaval rippmenüüst ja need on abiks kadreerimisel. Saadaval on kaheksat tüüpi kompositsioonijooni:

- Center

Jagab kaadri vertikaalselt ja horisontaalselt pooleks.

- Center Diagonal

Lisab diagonaaljooned.

- Thirds

Jagab kaadri horisontaalselt ja vertikaalselt kolmeks.

- Golden

Jagab kõrguse ja laiuse kuldloikelisteks osadeks (umbes 0,618 kaadrlaiuse kauguselt kaadri servast).

- Golden Triangle A

Joonistab alumisest vasemast nurgast ülemise parempoolseni diagonaali, seejärel lisab sellega 90 kraadi all ristuvad jooned, mis lõikuvad vastasnurkadega.

- Golden Triangle B

Sama nagu A, kuid vastasnurkadest.

- Harmonious Triangle A

Joonistab alumisest vasemast nurgast ülemise parempoolseni diagonaali, seejärel lisab sellega ristuvad jooned, mis lõikuvad ühest otsast 0,618 kaadrlaiuse kauguselt ja teisest otsast lõikuvad vastasnurkadega.

- Harmonious Triangle B

Sama nagu A, kuid vastasnurkadest.

Kaamera navigeerimine

Siit leiad mõned mugavad viisid kaamera paigutamiseks ja sellega navigeerimiseks.

NB!

Pea meeles, et aktiivne "kaamera" võib olla ükskõik milline objekt. Neid tegevusi saad kasutada ka näiteks valgusti paigutamiseks.

Kaamera vaatega kohakuti liigutamine

Mode: Objektirežiim (Object mode)

Hotkey: CtrlAlt0 NumPad

Selle klahvikombinatsiooniga saad kaamera tõsta aktiivse vaatega samasuunaliseks.

Vali kaamera ja paiguta 3D-vaade nii, nagu kaamerat paigutada soovid. Vajuta nüüd CtrlAlt0 NumPad; aktiivne kaamera paigutatakse automaatselt sinu vaatepunkti ja aktiivseks lülitatakse kaameravaade.

Kaamera paigutamine vaate abil

Kui kaameravaates olles lülitada omaduste paneelilt sisse Lock Camera to View, juhtub see, et kaameravaate muutmine liigutab ka stseenis olevat aktiivset kaamerat. 3D-vaates navigeerimine toimib nii nagu ennegi.


Roll, Pan, Dolly ja Track

Nende kaameraliigutuste tegemiseks peab kaamera olema valitud. Kaamera valimist saab teha ka kaameravaates olles. Selleks tuleb parempoolse hiirenupuga kaamera raamil klõpsata. Järgnevad tegevused eeldavad kaameravaates viibimist (0 NumPad)! Pärast vaate valimist saame kaamerat liigutada nagu igat teistki objekti:

Roll ehk keerutamine: kaldenurga muutmiseks vajuta klahvi R. Vaikimisi toimub keerutamine (*Roll*) ümber kohaliku Z-telje (kaamera vaatedelg), see on siis nagu fotoaparaadis portree ja maastiku pildistamise vahel valimine.

Vertical Pan ehk üles-alla suunamine: see on pööramine kohalikul X-teljel. Objekti pööramisrežiimi saamiseks vajuta klahvi R, siis vajuta kohaliku telje saamiseks kaks korda klahvi X. See toimib iga teljega. Täpsemalt vaata [telje lukustamise lehekülge](#)).

Horizontal Pan ehk vasemale-paremale suunamine: see tähendab kaamera pööramist ümber kohaliku Y-telje. Selleks vajuta R klahvi ja seejärel 2 korda Y klahvi!

Dolly ehk lähemale-kaugemale sõitmine: kaamera lähemale-kaugemale sõitmiseks vajuta klahvi G ja siis MMB , või kaks korda Z.

Vasemale-paremale sõiduks vajuta G, kaks korda X ja lohista hiirt. **Üles-alla sõiduks** vajuta G, kaks korda Y ja liiguta hiirt.

Kaamera lendamisrežiimis sihtimine

Kui oled kaameravaates, siis [lendamisrežiim](#) muudab ka kaamera asukohta...

[\[video link\]](#)

Perspektiiv renderdamisel

Kui vajutad F12 ja renderdusprotsess lõpetab, näed sa pilti kaamera "perspektiivvaates". Nii nagu objekti saab 3D-vaates erinevas perspektiivis vaadata, saab ka erinevas perspektiivis renderdada. Perspektiiv sõltub objekti vii suurusest ja kaugusest. Igal perspektiivvaatel on erinev arv koondumispunkte ehk pagupunkte.

Kui sa vaatad 3D-kuubiku pilti, näed kolme erinevat tüüpi servasid: vertikaalseid, horisontaalseid ja sügavuti. Kui kõik erineva tüübi servad on omavahel paralleelsed, siis pagupunkt puudub. Kui need pole paralleelsed, siis joonlauaga pikendamisel need lõikuvad mingis punktis. Selle punkti nimi ongi pagupunkt.

Eriliste vajaduste korral seatakse üles erineva perspektiiviga kaamerad. Mõningatel puhkudel tuleb pagupunktide arvu piirata, eriti arhitektuuriliste eesmärkide korral. Arhitektid ja joonestajad peavad joonisel tõepäraseid proportsioone näitama.

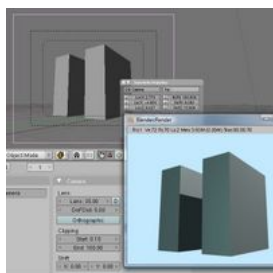
Näidisrenderdusi vaadates paistab hoone nagu mudast tehtud ja parajasti kokku kukkumas. Kui anda ehitajale korraldus selle järgi ehitada, oleksid ehitise seinad kaldu ja ülemised korrused järjest pisemad.

Vanas Kreekas ehitati sambaid, mis ülalt laienesid, nii et alt üles vaadates tundusid küljed paralleelsetena. Siis hakati sambaid ülalt kitsamaks suruma ja sambad tundusid tegelikust kõrgemadki. Paopunktide kasutamine kunstis arenes välja renessansiajastul. Blenderis on paar trikki sama saavutamiseks.

Märkus

Jätkamiseks on vaja tunda [kaamera seadistamist](#).

Kolmepunktiline renderdus



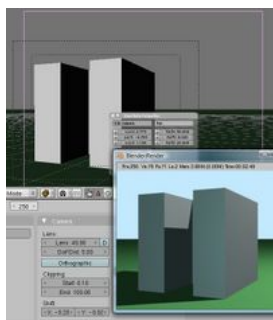
Tavaline kolme punkti renderdus

Palja silmaga vaadates või 35 mm kaameraga kõrget hoonet kerge nurga all alt üles pildistades saame kolme punkti perspektiivi. Kui asetad vertikaalsetele joontele joonlaua, märkad sa, et need koonduvad ühte punkti ehitise kohal.

Horisontaalsed jooned koonduvad ühele poole (vasakus näites) ja sügavusse taanduvad jooned koonduvad kusagile kolmandasse punkti (selle näite puhul kusagile alla paremale väljaspool pilti). Sellest tulebki nimetus kolmepunktiline renderdamine: on olemas kolm koondumispunkti.

Selline see reaalsus on ja viga pole siin midagi. Kõrghoonest välja astudes ja seda vaadates näed sa just seda. Sinu mõistus siiski teab, et ehitis on ruudukujuline ja sa muudad oma taju, nii et ehitis ei tundu pikali kukkuvat.

Kahepunktiline renderdus

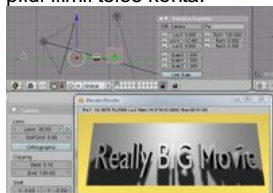


Kahe punktiga püstine renderdus

Tavalist arhitektuurilist renderdamist kutsutakse kahe punkti renderdamiseks; püstised jooned on paralleelsed, horisontaalsed koonduvad ühte punkti ning sügavusse kulgevad jooned teise punkti.

Arhitektidele selline renderdus meeldib, kuna siis on hoone püstised servad täiesti vertikaalsed ja hoone ei näi olevat ümberkukkumise äärel. Ka kompositsiooni mõttes ja skeemide tegemiseks on see variant päris hea, sest püstised jooned on paberi serva või ekraani servaga paralleelsed.

Varem tuli kahe punkti perspektiivi saamisega kaamera horisondile suunata. Siis on aga tulemuseks poole pealt "ära lõigatud" hoone ja pilt on igav. Arhitektuurifotograafid kasutavad selle probleemi lahendamiseks "nihutatud objektiivi". Objektiivi nihutamine paigutab pildi filmil teise kohta.



2-punktiline horisontaalne

renderdus

Seda on hea kasutada nii kõrgete objektide kui ka tava suurusega objektide puhul.

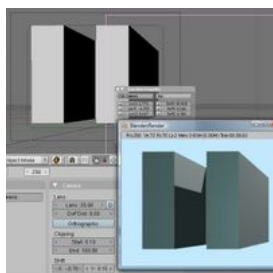
Enamikel juhtudest omavad koondumisjooni horisontaalsed ja sügavusjooned, vertikaalsed jooned on aga üksteise suhtes paralleelsed. Siiski tehakse mõningad pealkirjad nii, et horisontaalsed jooned on paralleelsed, kuid vertikaalsed ja sügavusjooned koonduvad. See suurendab tihti massiivsust ja kõrgust.

Antud efekti saamiseks aset kaamera tsentreeritult maapinna lähedale, suuna ülespoole ja nihuta renderduspaartuu alla. Toodud näites on kaamera 30 kraadi ülespoole suunatud ja asub pealkirjaga sama aluse kõrgusel. Ere, kiire kahanemisega juhtvalgus annab lavaliku välimuse ja võimendab samuti sügavust.

Kahe koondumispunktiga renderduse tegemiseks:

- Kasuta lainurkset 10 mm objektiivi hoonele hästi lähedal või siis suuremat objektiivi hoonest kaugemal. Suurema objektiiviga paistab ehitis saledam, vähem lavalik ja vähem moonutatud. Selles näites on kasutatud 40 mm objektiivi.
- Liiguta kaamera objekti suhtes rohkem ühe külje peale ja jät vertikaalselt hoone keskosa kõrgusele. Nii on hoone vertikaalsete joonte moonutus väike. Kaamera Z-väärtust võib suurendada, et see oleks aluspinnast või hoone ülaosast kõrgemal (kui on soov hoone katust näha). Ehitise eespool oleva alumise nurga esiletoomiseks tõsta kaamerat kõrgemale.
- Paiguta kaamera hoonest eemale, suuna täpselt horisondile, mitte üles ega alla (jälgi 20-kraadist Z-telje nurka näites). See peaks vertikaalsed jooned paralleelseteks muutma. Mida lähemal on kaamera objektile, seda lähemale tulevad horisontaalsete joonte koondumispunktid ja suureneb ka tunnetatav sügavus, sest lähemale liigub ka sügavusjoonte koondumispunkt.
- Kaamera võib kergelt allapoole suunata (1 kraadi kanti), et nii lähemal kui ka kaugemal olevad vertikaalsed jooned jookseksid ülalt alla. Kui jooned on siiski kõverad, kasuta pikemat objektiivi. Vertikaalseks saamiseks kasuta 3D-vaate akna kaamera vaates paartuud või monitori enda piksleid.
- Liigu kaameraga objekti suhtes edasi-tagasi, kuni objekt on renderdatud pildi serva lähedal ja õige suurusega.
- Muuda Shift: X ja Y väärtust, kuni objekti paigutus on korrigeeritud.

Ühe koondumispunktiga renderdamine



Ühe koondumispunktiga renderdamine

Ühe koondumispunktiga renderdamine on selline, et vertikaalsed ja horisontaalsed jooned on paralleelsed, aga sügavusjooned koonduvad. Selline renderdus meeldib arhitektidele, sest ettepoole paistvad küljed on õige küljesuhtega ja ruudukujulised, "külgmised" küljed on aga kaugusesse koonduvad.

Kui kaamera on paigutatud maapinna lähedale, näeb see välja väga lavalik, kuid samas veidi veider. Aeg-ajalt renderdatakse niimoodi tiitreid.

Ühe koondumispunktiga renderdamine

- Veel lavalikuma joone saamiseks kasuta lainurkset (10 mm), aga suure vaateväljaga objektiivi ja pane see ehitisele väga lähedale. Normaalsema mulje saavutamiseks jää 35 mm objektiivi juurde.
- Nihuta kaamera hoonest pisut paremale ja katusest pisut kõrgemale (juhul kui tahad, et katus nähtav oleks) või siis maapinna tasemele (näidispildil on kaamera peaaegu maapinna tasemel). Kui langetad kaamera allapoole maapinna taset, sulavad põhja jooned ja horisontaalsed jooned üheks ja tulemuseks on "väga dramaatiline" efekt.
- Suuna kaamera nii, et see vaataks täpselt taha ja oleks esikülje suhtes risti. Vertikaalsed jooned peaksid olema paralleelsed. Keera kaamerat Z-teljel kergelt objekti suunas, kuni horisontaalsed jooned saavad paralleelseteks. Tehniliselt võttes korrigeerid sa parallaksit (seda sõna on hea kasutada tüdrukule mulje avaldamiseks). Näidispildil on kaamerat objekti suhtes keeratud 0,5 kraadi.
- Nihuta kaamerat objekti suhtes edasi-tagasi, kuni see näib paartuu suhtes õige suurusega olevat.
- Muuda Shift: Y-väärtust seni, kuni paartuu alaosa (või tihti alajoon, kui on vaja pisut maapinda hoone ees näidata) on hoonega tasa. Muuda X-i seni, kuni ehitis on keskel (või siis kunstilistel kaalutlustel keskkohast eemal või siis hoopis selleks, et oleks näha parkimismaja), nagu on näidatud pildil.

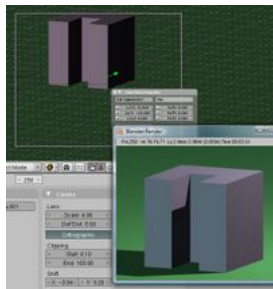
Näiteks toodud ekraanipildil on objektiivi suuruseks 35, X on negatiivne ja Y on positiivne. Kaamera asub objektist paremal sellega sama aluse kõrgusel. Kui X & Y oleksid nullid, oleks hoone paartuust väljas. Kusagil vasemal ülal.



Paralleelsed horisontaalsed servad

Horisontaalsete joonte saamiseks võid abijoonena paartuud kasutada. Muuda kaamera nurka seni, kuni horisontaalsed jooned paralleelseteks muutuvad.

Paralleelprojektsioonis renderdamine (*Orthographic Rendering*)



Paralleelprojektsioonis renderdamine

Ühegi punktita renderdamine on selline, kus nii vertikaalsed, horisontaalsed kui ka sügavusjooned on omavahel paralleelsed ja vaatenurk on tavaliselt 45, 30 või 60 kraadi. Kui kõik jooned on üksteise suhtes paralleelsed, puuduvad ka koondumispunktid.

Näidispilel on ehitis renderdatud nii, et kõik nurgad on 45 kraadi all. Nagu näha, on kõik vertikaalsed jooned omavahel paralleelsed. Samuti on omavahel paralleelsed horisontaalsed jooned ja ka sügavusjooned. Selliselt pildilt on lihtne välja lugeda, et hoone esimene ülaserv on sama pikk kui parempoolne ülaserv. Nii et hoone sügavus, laius ja kõrgus on samasugused; ka joonlauaga pilti üle mõõtes on tulemus sama.

Paralleelprojektsioonis renderdamine annab kuju edasi matemaatiliselt korrektselt. Paralleelprojektsioonis näed asju 3D-vaate aknas siis, kui sisse on lülitatud *View>Orthographic*.

Paralleelprojektsioonis renderduse saamiseks:

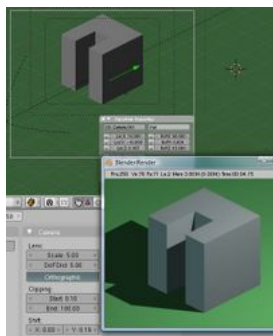
- Lülitka kaamera paneelil sisse Orthographic. <aegunud info>Nüüd on üks külg kaamera suhtes täiesti paralleelne</aegunud info>.
- Suuna kaamera objektile
- Paiguta kaamerat või suurenda vaadet nii, et objekt oleks soovitud suuruses

Läbi paralleelprojektsioonis kaamera vaatamisel pole objektiivi suurust enam olemas ja valguskiired ei koondu. Kõik valguskiired tulevad kaamera poole paralleelselt ja nii saab muuta ainult renderdatava ala suurust, mitte aga perspektiivi.

Määra *Shift X & Y* 0 peale ja kaamera objekti suhtes ülalt 45 kraadi all ja samas ajal ka külgvaate mõistes 45 kraadi all. Nii et kuubiku puhul näeksid kõik küljed geomeetriliselt ühesugused välja.

Paralleelprojektsioonis renderdused tehakse tavaliselt objekti suhtes 30, 45, või 60 kraadi all. Joonise lugeja saab mõõtmel teada trigonomeetria abil.

Isomeetriline renderdamine



Isomeetriline renderdamine

Kui me juba siin oleme, siis tuleb juttu ka isomeetrilisest renderdamisest, mis on paralleelperspektiivis renderdamise erijuht. Seda kasutatakse tihti joonestamisel ja "tegevust ülalt vaatavates" arvutimängudes.

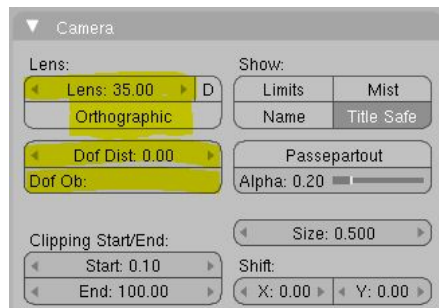
Isomeetrilise renderdamise puhul on vaja, et sügavusjooned oleksid horisontaalsete suhtes 30 kraadi ja et vertikaalsed jooned oleksid vertikaalsed. Täpsema selgituse saamiseks vaata [Wikipedia keerulist vektorarvutust](#). Isomeetrilise renderduse saamiseks:

- Lülitka kaamera paralleelprojektsioonile (*Orthographic*)
- Lisa kaamerale *Track To* piiraja (*constraint*), nii et *Target*-väljal on objekti nimi, mille poole kaamerat suunata. *To*: on -Z ja *Up* on Y.
- Paiguta kaamera nii, et see oleks objekti XY-telje suhtes 45 kraadi all ja siis 30 kraadi allapoole keeratud. Kui objekt on koordinaatidel 0,0,0, siis kaamera peaks olema koordinaatidel 10, -10, 10 või siis teist külge kasutades -10, -10, 10.
- Suurenda/vähenda kaamera vaatesuurstust nii, et objekt jääks paspartuust sissepoole.
- Kohanda väärtust *Shift: Y*, kuni renderdatav objekt on pildi keskel.

Sügavusteravus (*Depth Of Field, DOF*)

Päris kaameratesse ja sinu silma tuleb valgus läbi läätse, mis murrab valgust, ja läbi iirise (ava), mis piirab filmini, CCD/CMOS-sensori või silmapõhjani jõudva valguse hulka. Läätse ja iirise koostöö tõttu on nii, et mõned kindlal kaugusel olevad objektid on fookuses, ees- ja tagapool olevad aga fookusest väljas. Seda nimetame me sügavuskauguseks või siis "Z"-kauguseks kaamerast või silmast.

Pärismaailmas jõuab valgus läätсени mingi nurga all, mingist suunast. See mida näed, sõltub ka perspektiivist; lähemale liikudes paistab stseen teise nurga alt. Blender võimaldab ka "lamedate" piltide renderdamist, nagu seda kasutatakse arhitektuuriiliste jooniste puhul. Sellist kuvamise viisi nimetatakse paralleelprojektsiooniks (*Orthographic*). Seega on renderdamiseks kaks varianti, perspektiiv- ja paralleelprojektsioon. Perspektiivprojektsiooni puhul on pildil eespool olevad asjad suuremad ja tagapool väiksemad, kuni lõpuks ei ole väga kaugel olevaid asju üldse näha. Paralleelprojektsioonis (vaikimisi välja lülitatud) vaate puhul objekti asukoht selle kuvamise suurus ei mõjuta.



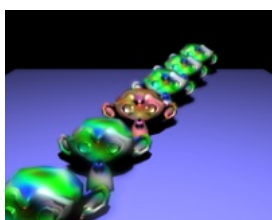
Iirise läbimõõdust sõltub, kui suures vahemikus on objektid veel fookuses. Kaamerates saab iirise (ava) suurus muutuda nupu alt *f-stop*. Teisisõnu öeldes on olemas "vaateväli". See on siis see mida sa ala, mida sa näed vasemalt paremale ja ülalt alla. Mingil kindlal kaugusvahemikul silmast on objektid fookuses. Näiteks öösel saad objekte fookuses hoida vahemikus 3 kuni 5 meetrit. Samal ajal on kolmest lähemal ja viiest kaugemal olevad asjad hägused. Sellisel puhul on sinu "sügavusteravus" 2 meetrit.

Mida suurem on iiris, seda väiksem on sügavusteravus. See on see, miks suudad päeval korraga fookuseerida suuremat piirkonda: väike iiris annab teravama pildi. Filmitööstuses on olemas selline ametimees, kes tegeleb näitleja nina ja kaamera vahemaa mõõtmisega, et oleks võimalik kaamera perfektelt fookuseerida.

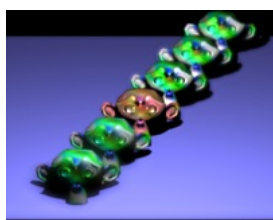
Mida kaugemal on objekt oma fookustasandist, seda hägusam see välja paistab. Seega on sügavusteravus see ala fookuspunkti ette- ja tahapoole, kus hägusus ei ole tajutav. Blenderis on selle kauguse nimetus Dof Dist või fookusteravuse kaugus (*Depth of Field Distance*) ja seda saab määrata omaduste paneelilt siis, kui kaamera on 3D-vaates valitud. Alternatiivne meetod fookuse hoidmiseks on panna Dof Ob väljale objekt, mille kaamera alati fookuses hoiab.

Vaateväli ja läätse fookuskaugus

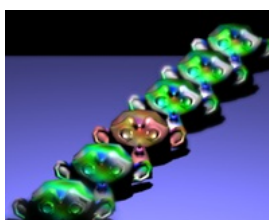
Nähtav ala sõltub objektiivi pikkusest. Kaamerates 35 mm on küllaltki tavapärane pikkus; sellega tehtud pilt sarnaneb silmaga nähtavale ja pildistada saab küllaltki lähedalt. Läätse fookuskauguse muutmiseks kasuta kaamera seadistuste paneeli (vaikeväärtus on 35 mm). Pikem objektiiv veidi kaugemal saab sama ala teise perspektiiviga näidata. See meeldib paljudele režissööridele, kuna see ühtlustab pildi dünaamikat, kuna kaamera on tegevusest palju kaugemal.



35 mm objektiiv 10 ühiku kaugusel.



210 mm objektiiv 60 ühiku kaugusel samas kohas/pöördel.



210 mm objektiiv 50 ühiku kauguselt; ümberpaigutatuna nii, et ala oleks 35 mm objektiivi sarnane.

Suurendamine Blenderis

Suurendamine tähendab pildi mingi osa laiendamist kogu nähtavale alale; meil inimestel selline võime puudub. Ma pean siiski sõnad tagasi võtma; tegelikult suudame: tuleb lihtsalt diivanilt tõusta ja lähemale kõndida (kuigi see on rohkem küll "rändamine" kui "suurendamine"). Blenderiga saab teha mõlemat: võib kaameraga "rööbastel" (või "kärul") edasi-tagasi liikuda või võib ka muuta objektiivi suurus. Seda tegevust võib juhtida ka lpo-kõveratega.

Sügavusteravus arvutigraafikas

Arvutigraafika puhul füüsilised objektiivid ja avad puuduvad. Seetõttu on kõik objektid alati fookuses, sügavusteravus lõpmatu. Kunstilistel põhjustel soovime me tihti, et esiplaanil olevad tegelased oleks fookuses ja taustal olev hägune, sest siis ei pööra vaatajad (ega kunstnik) taustale liigset tähelepanu. Fokuseerimisega on võimalik ka peategelasi kõrvaltegelastega võrreldes esile tuua. Seega tuleb realistliku sügavusteravuse illusioon komposiitoris järeltöötuse sammuna ise luua.

Renderdamine ja piltide salvestamine

Pärast renderdamise seadistamist [kvaliteedi](#) ja [tulemuse](#) järgi, on renderdamine ise juba üsna lihtne. [Animatsiooni renderdamine](#) on pisut keerulisem ja sellest on juttu järgmises osas.

Aktiivse kaamera pildi renderdamiseks tuleb paneelil Render vajutada suurt nuppu Image. Vaikimisi asetub selle peale 3D-vaate aken UV/Pildiredaktori aknaga ja renderdamine algab.

Renderdatud pildi kuvamine

Jooksvat renderdamist näidatakse Pildiredaktori aknas. Menüüst Display saab valida erinevate esitusviiside vahel:

Keep UI (säilita kasutajaliides)

Renderdatud pilt läheb Pildiredaktorisse, aga kasutajaliides ei muutu. Renderdatud pildi nägemiseks tuleb ise Pildiredaktorisse minna.

New Window (uus aken)

Renderdatud pilti kuvatakse eraldiseisvas aknas

Image Editor (pildiredaktor)

3D-vaade asendatakse renderdatud pilti kuvava Pildiredaktori aknaga

Full Screen (täisekraan)

Kasutajaliides asendatakse kogu ekraani hõlmava ja renderdatud pilti kuvava Pildiredaktori aknaga

Iga valiku puhul saab klahviga Esc renderdatud pildi eest ära ja endise vaate tagasi.

Salvestamine

Renderdatud pildi saab salvestada, valides Menüüst *Image* valik *Save As Image*.

Esitusviisid

Renderdatud pildi kuvamisel Pildiredaktori aknas ilmuvad nähtavale mitmed lisamenüüd.

Slot Menu (pesad)

Kui renderdamise järel valida Menüüst Slot uus renderdamise pesa, renderdatakse järgmine pilt sinna. Renderdamispeassa renderdatud pilte saab hiljem vaadata ja võrrelda Slot-rippmenüüst pesasid vahetades. Tühje pesi kuvatakse tühjade ruudustikena. Renderduspesade vahel liikumiseks võib kasutada klahve J ning AltJ.

Render Layer (renderduskiht)

[Renderduskihtide \(Render Layers\)](#) kasutamisel saab siit Menüüst kuvatava kihi valida.

Render Pass (renderduskäik)

[Renderduskäikude \(Render Passes\)](#) kasutamisel saab siit Menüüst valida kuvatava käigu.

Image Painting (piltide maalimine)

Selle ikooni abil saab sisse lülitada piltide maalimise.

Display Mode (esitusviis)

Viimase nelja nupu abil saab muuta piltide kuvamise viisi.

RGB

Näita renderdatud pilti nii nagu on, ilma läbipaistvuskanalita.

RGBA

Selle puhul asendatakse pikslite läbipaistvus taustaruudustikuga.

Alpha Channel (alfakanal)

Kuvab pilti halltoonidena. Mustad alad näitavad, et need on läbipaistvad, valged aga, et läbipaistmatud.

Z Depth (Z-sügavus)

Näitab pikslite kaugust kaamerast ja lisaks nägemisulatuse algust ja lõppu, nagu on kirjeldatud peatükis [Kaamera seaded](#).

Curves Panel (kõverad)

Paneel Curves on leitav Omaduste paneelilt (Menüüst *ViewProperties*). Selle abil võid pildi värve kohendada.

Renderduse kvaliteet

Kvaliteeti mõjutavad mitmed tegurid. Stseeni renderdamine ilma renderdusseadeid muutmata annab suure tõenäosusega üsna inetu pildi. Eelmistes osades õppisid sa modelleerimist, varjutamist, tekstuurimist ja valgustamist. Nende optimeerimisega saab pildi kvaliteedi osas nii mõndagi ära teha, kuid enne render-nupule vajutamist on siiski vaja teha olulisi seadistusi. Need mõjutavad renderdatavat tulemust otseselt.

Järgnev peatükk katab renderduskihid ja renderduskäigud, mis mõlemad lubavad stseeni elemente kombineerida. Mõnikord on vaja efektid kohe renderdajast kätte saada, mitte aga neid järeltöötlusega tekitada.

Värvide haldamine (*Color Management*)

Üks tähtis 3D-renderduse aspekt, mis sageli jäetakse kahe silma vahele, on värvide haldamine. Ilma selleta näitavad renderdusmootorid küll stseeni valgust õigesti, kuid esitlevad seda sinu monitoril valesti. Blender lihtsustab seda protsessi, kuid on oluline teada, kuidas renderdatud pildi värviruum sinu töö puhul käitub.

Sakisilumine (*Antialiasing*)

Sakisilumine kaotab kontrastsete värvüleminekute juures tekkivad astmelised servad. See valik mõjutab renderduskvaliteeti suurel määral. Sakisilumiseta näevad pildid tavaliselt välja väga arvutilikud ja amatöörlikud.

Säriaeg (*Exposure*)

Säriaeg on füüsikaline termin ja see näitab kestvust, mille jooksul sensorit ühe kaadri saamiseks valgustatakse. Mida pikem säriaeg, seda heledam pilt. Arvutigraafikas toimub säriaja simuleerimine valguse heleduste nihutamise abil.

Sügavusteravus (*Depth of Field*)

Päris kaameratel puudub konkreetne fookuskaugus. See on see vahemik, milles olev on fookuses ja terav. Fookuses oleku või fookusest väljas oleku määraavad ära mitmed tegurid. Vaikeseadistustega renderdades on kõik alati fookuses. Sügavusteravusega mängides saab luua näiteks ettekujutuse objekti ligikaudsest suurusest.

Liikumishägu (*Motion Blur*)

Kaamerate puhul saab muuta säriaega ehk filmi valgustamise aega. Valgustamise ajal liikuvad objektid jätavad häguse pildi. Kiiremini liikuvad objektid on hägusemad kui aeglaselt liikuvad objektid. See on arvutigraafikas oluline efekt, sest me eeldame selle kõrvalmõju olemasolu. Vastasel korral ei ole pilt usutav.

Sakisilumine (*Antialiasing*)

Arvutiga tehtud pilt saab koosneda ainult pikslitest, millest igaüks saab olla vaid üht kindlat värvi. Renderdamismootor annab igale pikslile värvi, sõltuvalt sellest, mis objekt sellel pikslil näha on. Kuna inimese silma lahutusvõime on suurem kui ekraanil, paistavad taustast tugevalt eristuvate objektide servad ja peened jooned sakilistena.

Sakilisuse probleemi saab kõrvaldada, kasutades sakisilumist (*Anti-Aliasing*). Põhimõte seisneb selles, et iga piksli väärtust arvutatakse väikese nihkega mitmekordselt, näiteks 5 või rohkem korda. Seejärel antakse lõplikule pikslile jaotatud osade keskmine värv.

Sakisilumise ehk teisisõnu lisadiskreetimisega (*Oversampling, OSA*) seotud nupud asuvad renderduspaneeli alampaneelis Render.

Valikud

Linnuke Anti-Aliasing
ees lülitab sakisilumise sisse

5 / 8 / 11 / 16

See number näitab, kui mitmeks osaks üks piksel jaotatakse. 5, 8, 11, ja 16 on sisseehitatud numbrid, igaühel neist on oma kindel paiknemismuster. Suurema arvu korral renderdusaeg kasvab, kuid servad on ilusamad.

Vaikimisi kasutatakse Blenderis fikseeritud ühtlase nihkega tabelit. Piksli osad jagatakse nii, et kindlasti oleks:

1. Võrdne sãmplite omavaheline kaugus naabersãmplitest
2. Vertikaal- ja horisontaalsuundade alampiksliid ühtlaselt kaetud

Allolevad pildid näitavad Blenderi mustreid 5, 8, 11, ja 16 korral. Et näidata samasugusust kõigi pikslite peal, on näidatud ka naaberpiksliid. Iga piksli puhul kasutatakse sama mustrit.

5 sãmplit

8 sãmplit

11 sãmplit

16 sãmplit

Full Sample (säilita alamosad)

Kõik sakisilumise alamsammud salvestatakse renderduskihtide sisendiks. Selle valiku abil saab ära hoida komposiitmisel tekkivad sakisilumise probleemid.

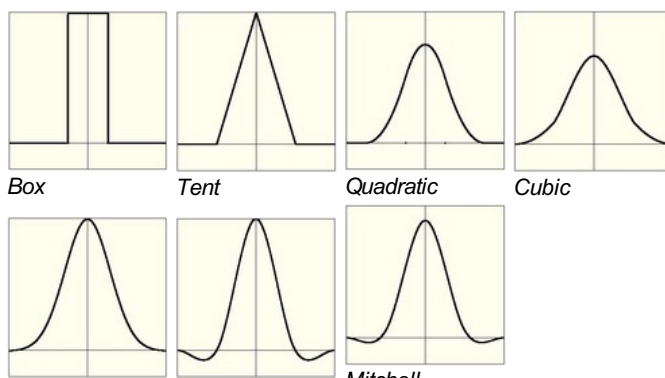
Filtreerimine

Pärast renderdamist on iga alampiksli kohta olemas info värvi ja läbipaistvuse kohta. On oluline määrata, kui palju igaüks neist lõpp-piksli mõjutab.

Lihtsaim variant on kõigi piksli segmentide keskmistamine ja saadud keskmisega piksli värvi määramine. Selle nimetus on ruutfilter (*Box Filter*). Selle variandi puuduseks on aga see, et mõned pikslisegmendid on piksli servale väga lähedal ja mõjutavad selle tõttu ka naaberpiksleid.

Filtrimenüü: Saab määrata, kuidas saadakse alampikslite "keskmine":

Box (ruut)	Selle filtriga saab suhteliselt kehvast kvaliteedist. Varasemates Blenderi versioonides oli see vaikevalik. Ruutfilteri puhul kasutatakse ainult piksli sees olevaid segmente. Teiste variantide puhul kasutatakse ka naaberpikslite segmente.
Tent (telk)	Lihtne filter, mis annab terava tulemuse
Quadratic (kvadraat)	Neljanda astme kõver
Cubic (kuup)	Kuupfunktsiooni kõver
Gauss	Gaussi kõver, kõige hõõgus
Catmull-Rom	Catmull-Romi filter, kõige teravam
Mitchell-Netravali	Mitchell-Netravali, hea ja paraja teravusega filter



Gaussian

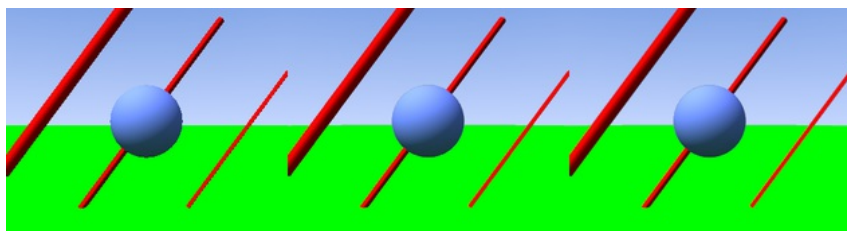
Catmull-Rom

Mitchell-
Netravali

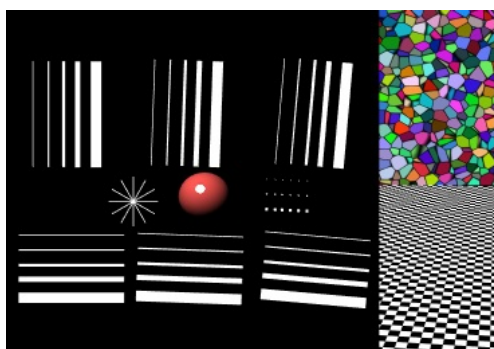
Filtri suurus

Selle numbri vähendamine koondab võetavate sümplite asukoha piksli keskkoha lähedale ja nii tekib hägusem pilt. Suurem number teeb tulemuse teravamaks. Kahel viimasel filtril on ka negatiivne külg: on võimalik liigse teravdamise efekt.

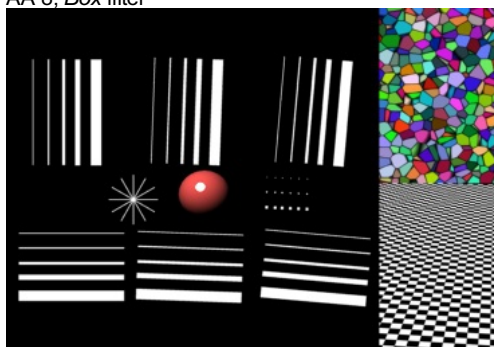
Näited



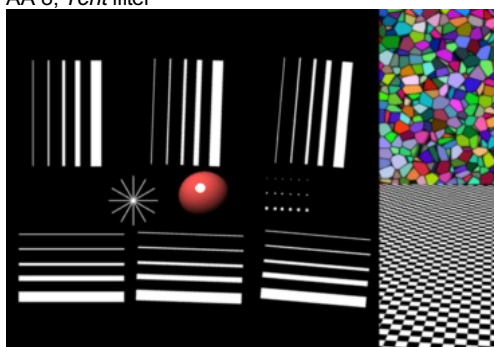
Ilma sakisilumiseta (vasemal), koos sakisilumisega AA=5 (keskel) ja AA=8 (paremal).



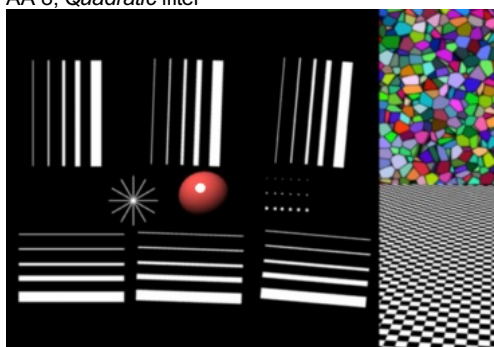
AA 8, *Box* filter



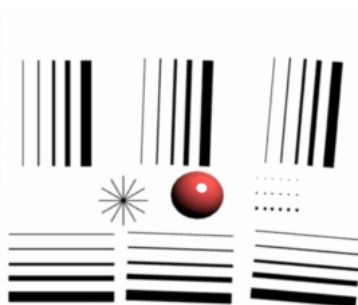
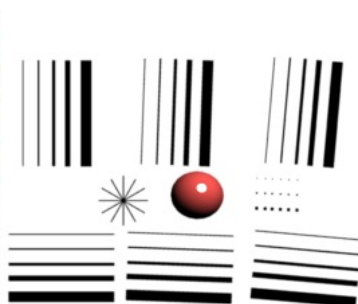
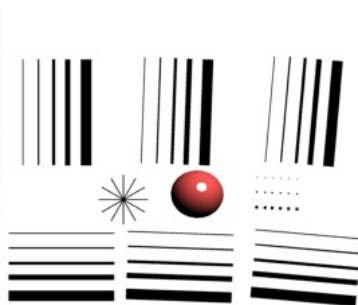
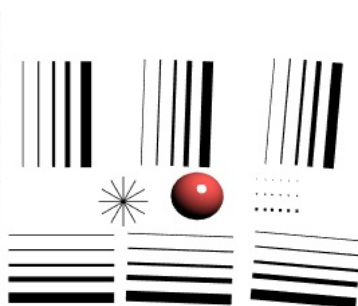
AA 8, *Tent* filter

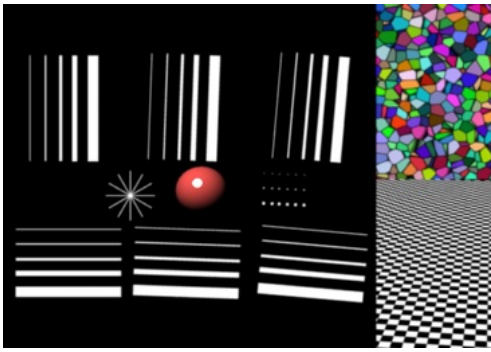
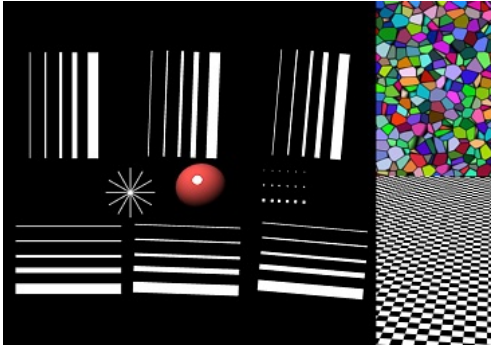
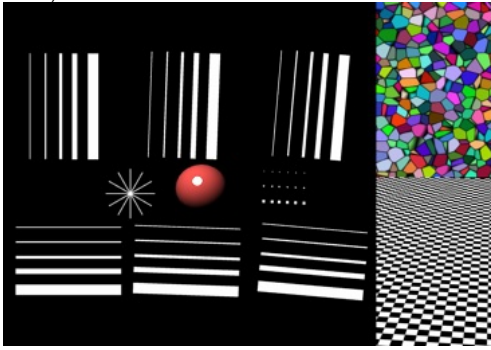
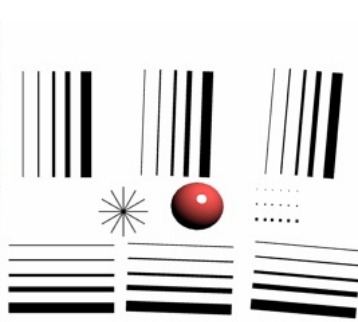
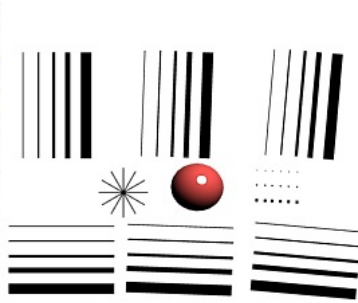
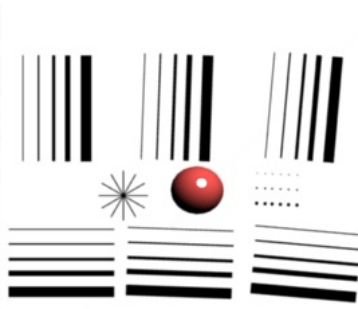


AA 8, *Quadratic* filter



AA 8, *Cubic* filter



AA 8, *Gaussian* filterAA 8, *Catmull-Rom* filterAA 8, *Mitchell-Netraval* filter

Animatsioonide renderdamine

Üksiku pildi renderdamisel saab seda mälupuhvril vaadata ja alles siis kettale salvestada. Animatsiooni puhul salvestatakse aga pildid kohe pärast renderdamist kettale.

Pärast renderdamist võid soovida videoklippe monteerida, roheekraani (*green screen*) maskimiseks komposiitorit kasutada, lisada taustu (*matte*), korrigeerida värve, seadistada fookuskaugust või midagi muud. Väljundi saab suunata monteeri jasse (*Sequencer*), kus klipid lõigutakse, järjestatakse ja kihindatakse.

Lõpuks saad monteeri jast lõpptulemuse välja kirjutada ja sellest esitluskõlbliku videofaili kokku pakkida.

Töövoog

Tavaliselt tehakse palju proovirenderdursi, enne kui ajastus, valgustus, paigutus, materjalid ja muu paika saab. Mingil hetkel jõutakse sinnamaani, et saab renderdada lõpliku animatsiooni.

Renderdada saab kaht moodi, sõltumata sellest, kas tehakse filmi, animatsiooni, helindatud või helita variandina. Lähenedistüüp tuleb valida vastavalt saadaolevale protsessoriajale (vaata [Renderdusjõudlus](#)). Lõpliku renderdusaja ligikaudseks saamiseks võid kõigepealt renderdada üksiku "tüüpilise" kaadri, lugeda kokku kaadrite hulga ja siis need omavahel korrutada.

Kui renderdamisele kulub aeg on tund või rohkem, on mõistlik renderdada kaadrite kaupa. Näide: kui ühes minutipikkuses videoklipis on kaadrisagedus 24, siis kokku 60 sekundit \times 24 kaadrit = 1440 kaadrit. Kui iga kaadri renderdamiseks kulub 30 sekundit, tähendab see 2 kaadrit minutis. Kokku on renderdamiseks vaja 720 minutit ehk 12 tundi.

Renderdamine kasutab kogu saadaolevat arvutusvõimsust. Arvuti võiks renderdama panna ööseks. Võib teha ka nii, et jätta muude tööde tegemise ajaks Blenderile renderdamiseks vähem protsessori kasutusõigust. Jälgi ka kasutatava mälu kogust!

Ei ole soovitatav kasutada "otsest lähenedist". See oleks otse Blenderist AVI, MOV või muusse video konteinerformaati renderdamine. Blender tekitab kõiki kaadreid tervikuks ühendava faili. Võid kasutada heli lisamiseks ja MPEG-formaati salvestamiseks Blenderi enda videoredaktorit.

"Kaadrite kaupa" renderdamine on animatsiooni renderdamiseks palju kindlam viis. Selleks vali väljundiks pildiformaat (näiteks JPG, PNG või MultiLayer) ja vajuta nuppu ANIM.

Blender teeb iga kaadri jaoks eraldi pildifaili. Pärast seda võid järeltöötluseks (*post processing*) kasutada Blenderi komposiitorit. Seejärel võid saadud kaadrid tuua videoredaktorisse, lisada heli ja lõpptulemuse MPEG-videoks renderdada. Kaadrite kaupa renderdamine on keerukam ja kulutab rohkem kettaruumi. Samas annab see juurde paindlikkust.

Järgnevad mõned valiku langetamist hõlbustavad põhimõtted.

Otse videoks renderdamine

- lühikesed segmendid, mille renderdamiseks kulub maksimaalselt < 1 tund
- usaldusväärne elektrivool
- arvuti pole muudeks tegevusteks vaja kasutada

Kaadrite kaupa renderdamine

- renderdamisele kuluv aeg > 1 tund
- järeltöötluse vajadus
 - Värvuse/valgustatuse muutmine
 - Roheekraani/tausta muutmine
 - Kihindamine/komposiitimine
 - Mitmes formaadis ja erineva resolutsiooniga lõppvideod
- Koodek vajab kindlat kaadrisagedust
- täpne ajastus (näiteks hülte sünkroniseerimine heliga) mõningates kohtades
- vajadus renderdamist muu töö tegemise ajaks katkestada ja hiljem katkestatud koht pealt jätkata.

Kaadrite kaupa renderdamine

1. Kõigepealt valmista oma animatsioon ette.
2. Renderdamise resolutsiooni, küljesuhte, renderdatavate kaadrite vahemiku ja kaadrisageduse saad valida paneelilt Dimensions.
3. Väljundformaadiks vali mõni selline pilditüüp, mis ei vähenda kvaliteeti (eelistan kasutada PNG-d või MultiLayerit, sest need on kvaliteedikadudeta formaadid).
4. Vali väljundi kataloog ja trüki väljundi paneelile näiteks "render\my-anim"
5. Kontrolli renderdatava animatsiooni algus- ja lõpukaadrid (*Start* ja *End*)
6. Salvesta oma .blend-fail.
7. Vajuta suurt *Animation* nuppu Võta ette mõni pikem tegevus [nagu näiteks magamine, telekamäng, majaesise koristamine], kuni arvuti renderdamisega ühele poole saab.
8. Pärast renderdamise lõppu mine failihalduriga väljundkausta (Selles näites .render). Näed hulka pildifaili (.png või .exr vms, olenevalt sellest, mida renderdamisele saatsid), mille järjekorranumber on vahemikus 0000 kuni 9999. Need on üksikud kaadrid.
9. Blenderis mine nüüd [videomonteeri jasse](#) (Video Sequence Editor).
10. Vali lisamismenüüst (*Add*) käsk *Add Image* (Lisa pilt). Vali kõik animatsioonis kasutatavad kaadrid (kõigi valimiseks vajuta klaviatuuri A-klahvi). Need lisatakse ribadena (*strip*) monteeri jasse. Kui tavaliselt nimetatakse videomontaaži programmis monteeri tavaid video- ja helilõike klippideks (*clips*), siis Blenderis nimetatakse neid ribadeks.
11. Nüüd võid monteeri da ribasid, lisada efekte või jätta ribad nii, nagu on. Saad lisada ka teisi ribasid, näiteks heliribasid.
12. Kõigi vajalike kaadrite olemasolus veendumiseks lohista hiire vasemat klahvi all hoides ajatelg läbi.
13. Renderdamise alampaneelil Post Processing lülita sisse Sequencer.
14. Vali väljundformaadi paneelil sobiv koodek (näiteks MPEG H.264) ja seadista see. Videokoodekeid kirjeldatakse eelmisel

leheküljel: [Väljundformaadi valikud](#)

15. Vajuta Animation-nuppu ja Blender renderdab monteerijas tehtu videoks.

Mis seda nii keeruliselt teha? Esiteks saab kaadrite kaupa renderdades renderdamise alati Esc-klahvi abil katkestada. Selle hetkeni renderdatud kaadrid jäävad alles, sest need on salvestatud eraldi failidesse. Alati on võimalik muuta renderdusvahemikku ja jätkata katkestatud koha pealt.

Vahepeal on võimalik teha järeltöötlust. Monteerija aknas saab lisada mitmesuguseid efekte. Monteerijast saad sama faili kiiresti erineva resolutsiooni (640×480, 320×240 jne) ja koodekiga välja kirjutada.

Valikud

Väljundi paneel *Output*

Vaikimisi renderdatakse animatsioonid kausta, mis on määratud paneelil Output (Animation location and extensions - Animatsiooni asukoht ja faililaiendid). AVI-väljundvormadi puhul pannakse nimeks #####_#####.avi, kus '#####' näitab animatsiooni algus- ja lõpukaadrit 4-kohalise numbrina. Vajadusel lisatakse ette nullid.

Piltideks renderdamisel tekitatakse seeria pilte, mille nimeks on kaadri number.

File Extensions (faililaiendid)

Lisab failinimele ka õige faililaiendi

Overwrite (kirjuta üle)

Kirjutab juba varem renderdatud failid üle.

Placeholders (asukohahoidjad)

Tekitab enne renderdama asumist järjehoidjaks tühjad failid.

Järeltöötluspaneel *Post Processing*

Sequencer (monteerija)

Renderdab 3D-stseenis oleva kaamera asemel monteerija väljundi. Kui monteerija sisaldab stseeniribasid, renderdatakse ka need ühe osana. Kui komposiitor on sisse lülitatud, saab komposiitori väljundit kasutada monteerijaribadena.

Compositing (komposiitimine)

Saadab väljundi komposiitorisse, siis saadab pildid läbi selle sõlmestiku ning näitab pilti, mis jõuab väljundsõlme Composite Output

Vihjed

Oeh! Mu väike öde lülitas arvuti täpselt keset filmi renderdamist välja!

Kui animatsioon on vähegi keerukam ja renderdamisele kulub üle poole tunni, on soovitatav see renderdada ühe video asemel piltide jadana. Näiteks mõnda kadudeta formaati (TGA, PNG, BMP). Nii saab katkenud renderdamise korral katkenud kohast jätkata või siis ainult problemaatilised kaadrid uuesti renderdada. Lihtsalt muuda kaadrit Start, millest alates uuesti renderdada, ja vajuta uuesti ANIM.

Mul on vaja renderdada ainult mõned kaadrid klipi keskelt

Järjestatud kaadritesse renderdamine on hea selle poolest, et ainult osade kaadrite parandamise korral ei pea kõiki kaadreid uuesti renderdama. Kaadrite videofailiks kokkuliitmiseks võib kasutada Blenderi monteerijat või komposiitorit.

Renderdub vaid esimene kaader, siis jookseb Blender kokku

Enne ANIMATION-nupule vajutamist veendu, et väljundfail poleks mõnes esitavas programmis avamise tõttu lukustunud.

Renderdamise ajal tasuks jälgida konsoliakent.

Veateade

Unable to create Quicktime movie: CreateMovieFile error: -47

QuickTime'i videoriba on kasutuses (oletatavasti VSE-s (videomonteerijasse)) ja seda ei saa üle kirjutada. Kui riba on monteerijas, kustuta see riba või siis kustuta fail failihalduris.

Eelrenderdus

Eelrenderdamine (*baking*) on mõnede parameetrite ettearvutamine selleks, et mõnd hilisemat protsessi kiirendada. Nullist renderdamine võib sõltuvalt valikutest kaua aega võtta. Eelrenderdamisega saab valitud objektidel mõningad osad enne "päris" renderdamist ära renderdada. Sellisel juhul toimub pärast Render-nupule vajutamist renderdamine kiiremini, sest enam pole tarvis näiteks värvi uuestiarvutamist.

Eelrenderdamine loob objektide renderdatavast pinnast kahemõõtmelised pildifailid. Need laotatakse UV-koordinaate kasutades uuesti objektile peale. Eelrenderdust tehakse eraldi igale võrele ja seda saab teha vaid siis, kui võrel on UV-lahtilõige (*UV-unwrapping*). Kuigi selle ülesseadmine on aeganõudev, hoiab see renderdusaega kokku. Kui võrrelda eelrenderdamata renderdamist ja eelrenderdatud renderdamist, on säästetud aeg kindlasti suurem kui eelrenderdamise peale kulunud aeg.

Eelrenderdust tasub teha intensiivsetes valguse/varju olukordades, nagu näiteks kaudvarju (*Ambient Occlusion*) või alavalgustite pehme varju puhul. Kaudvarju eelrenderdamisel tuleb see täisrenderduse tegemisel välja lülitada. See säästab aega.

Tekstuuri loomiseks vali Full Render või Textures; eelrenderdatud protseduurilised tekstuurid võivad olla edaspidise tekstuuri joonistamise aluspildiks. Madala tihedusega võre kuvamiseks kõrgresolutsioonilisena vali Normals. Selle jaoks tee kõrgresolutsioonilisele võrele UV-laotus ja siis eelrenderda selle normaaliid. Salvesta see normaalikaart ja kasuta seda analoogselt lahtilõigatud madalresolutsiooniga võrel. Lõpuks näeb madalresolutsiooniline võre välja täpselt nagu kõrgresolutsiooniline, kuid sellel on tunduvalt vähem külgi.

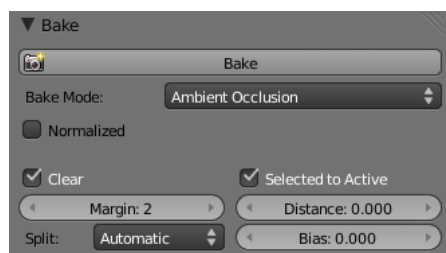
Plussid

- Võib renderdusaega oluliselt kahandada
- Tekstuuri maalimine on lihtsam
- Väheneb hulk tahukate arv
- Korduvrenderduste puhul on võidud veelgi suuremad

Miinused

- Objektile tuleb teha UV-laotus.
- Varjude eelrenderdamise puhul ei saa objekte ja valgusteid enam üksteise suhtes liigutada.
- Suured (näiteks 4096×4096) eelrenderdatud tekstuurid võtavad palju mälu ja võivad olla aeglasemad kui renderdamine.
- UV-laotuse, eelrenderdamise ja tekstuuri uuesti kanalile määramise jaoks kulub inimeste tööaega.

Valikud



Ambient Occlusion

Eelrenderdamise viis

Full Bake (täielik renderdus)

Eelrenderdab kõik materjalid, tekstuurid ja valgustuse, välja arvatud läige ja pinnasisene hajumine.

Ambient Occlusion (hajusvalgustus)

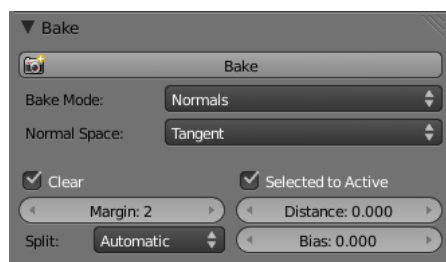
Eelrenderdab paneelil World määratud seadistustega kaudvarju. Ignoreerib kõiki stseeni valgusteid.

Normalized (normaliseeritud)

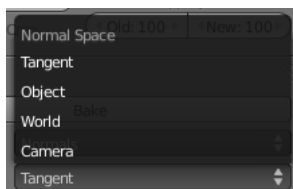
Normaliseeri materjali seadistusi kasutamata.

Shadow (vari)

Eelrenderdab varjud ja valgustuse.



Normaaliid (Normals)



Normaaliruum (Normal Space)

Normals (normaalid)

Eelrenderdab lõikuvad ja kaamera poolt nähtavad (kõigi muude hulgast) normaalid RGB pildiks.

Normal Space (normaaliruum)

Normaale võib renderdada erineva koordinaadistikuga:

Camera space (kaamerast lähtuv)

Vaikemeetod.

World space (maailmast lähtuv)

Normaalid eelrenderdatakse maailma koordinaatide järgi. Tulemus sõltub ka teisendustest ja deformatsioonidest.

Object space (objektist lähtuv)

Normaalid eelrenderdatakse objekti enda koordinaatides, sõltumata teisendustest, samas sõltudes deformatsioonidest.

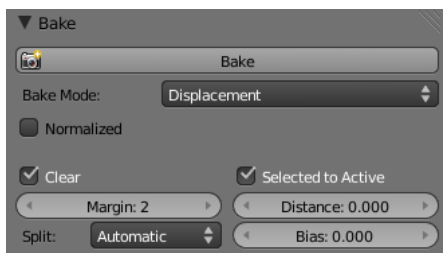
Tangent space (tangentsiaalruumis)

Normaalid objekti enda koordinaatides, sõltumata transformatsioonidest ja deformatsioonidest. See on uus vaikemeetod ja enamjaolt ka õige valik, sest normaalilaotust saab kasutada ka animeeritud objektide puhul.

Ka materjalide jaoks saab valida erineva ruumikoordinaatide kasutuse. See on pilditekstuuride valiku all valikute Normal Map kõrval. Korrekse tulemise saamiseks peavad seal tehtud valikud olema samad, mida kasutati eelrenderdamisel.

Textures (tekstuurid)

Eelrenderdab ainult ilma varjutuseta materjalid ja tekstuurid.



Displacement

Displacement (nihutus)

See on samane normaalkaardile. Nihutuskarte saab samuti detailsetest objektidest eelrenderdada ja siis neid madala resolutsiooniga objektide peal kasutada.

Normalized (normaliseeritud)

Normaliseeri kauguse järgi.

Pinnatükelduse- ja nihutamistöötlejaga kasutamisel tuleb enne madalresolutsioonilis mudeli eelrenderdamist sellele ajutiselt lisada tugev pinnatükelduse töötleja (Subsurf). See tähendab, et nihutamistöötleja kasutamisel pinnatükeldaja kohal on nihutamine korrektne, sest see talletatakse suhtelise erinevusena tükeldatud pindade geomeetrias. Mida suurem on renderdustaseme tükeldusaste eelrenderdamise ajal, seda täpsem on nihutus. Sellist tehnikat võib kasutada ka siis, kui on plaanis nihutamiskaarti välise renderdaja jaoks eksportida.

Emission (valguskiirgus)

Eelrenderdab materjali valguskiirguvuse või kuma.

Alpha (alfa)

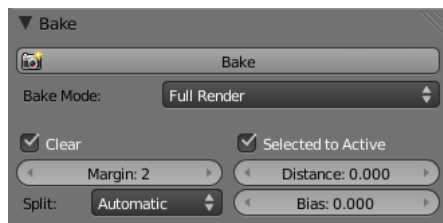
Eelrenderdab materjali läbipaistvuse.

Mirror Color and Intensity (peegli värv ja intensiivsus)

Eelrenderdab peegeldusel muutunud värvi ja intensiivsuse.

Specular Color and Intensity (läike värv ja intensiivsus)

Eelrenderdab läikevärvi ja intensiivsuse.



Täielik renderdus (Full Render)

Lisavalikud

Clear (puhasta)

See valik teeb enne renderdamist pildi tühjaks, värvides selle üleni taustavärviga (see on vaikimisi must).

Margin (ääris)

Jätab igale UV "saarekesele" määratud pikslite jagu lõikevaru, mis tagab sujuvamad lõikeservad.

Split (jaotamine)

Fixed (ettemääratud)

Jaga nelinurgad ettemääratult (0,1,2) (0,2,3).

Fixed alternate (alternatiivne ettemääratud)

Jaga nelinurgad ettemääratult (1,2,3) (1,3,0).

Automatic (automaatne)

Jaga nelinurgad nii, et eelrenderdamisel oleks kõige vähem moonutusi.

Select to Active

Võimalda teiste objektide informatsiooni eelrenderdamist aktiivse objekti sisse.

Distance (kaugus)

Määrab, kui kaugel võib aktiivse objekti punkt olla teise objekti punktist. Tarvilik vaid siis, kui valik Selected to Active on aktiivne.

Tavaline kasutusstsenaarium on kõigepealt luua suure punktitihtedusega objekt ja siis eelrenderdada selle normaalid madala punktitihtedusega objektile. Saadud normaalilaotust võib seejärel kasutada *low-poly* objekti juures, sellele detailide lisamiseks.

Bias (määr)

Määra, kui kaugel olevaid objekte kaasatakse (Blenderi ühikutes)

Võre peab renderdamise jaoks nähtav olema.

Kui objekt on ülevaateaknas (Outliner Window) renderdamise jaoks välja lülitatud, ei ole eelrenderdus võimalik.

Töövoog

1. Aktiveeri 3D-vaates soovitud võre ja klõpsa muutmisrežimis 3D-vaate päises olevat külgede valimise nuppu Face select mode.
2. [Loo võrele UV-laotus](#)
3. Tee aknas UV/Image Editor uus pilt või kasuta olemasolevat. Kui 3D-vaade on tekstuuri kuvamise režimis, peaksid nägema pilti võrele laotatuna. Veendu, et kõik küljed oleks valitud.
4. Vali renderdamiskonteksti paneelilt Bake soovitud pildi eelrenderdamise tüüp (näiteks Full Render) ja tee eelrenderdus.
5. Pärast eelrenderdamist asendab Blender valitud pildi eelrenderdatuga.
6. Salvesta pilt.
7. Kasuta saadud pilti UV-tekstuurina. Normaalitekstuuri ja nihketekstuuri kasutamiseks vaata peatükki [Normaali- ja nihketekstuurid](#). Täieliku eelrenderdamise ja tekstuuri eelrenderdamise kohta vaata peatükki [Tekstuurid](#).
8. Viimistle pilti eeltoodu abil, kasuta selleks [Tekstuuri maalimise režiimi](#) või eraldi pilditöötlusprogrammi.

Windowsi kasutajad eelrenderdavad kõigepealt hajusvalgustuse (Ambient Occlusion).

Blenderi kasutamisel Windowsis tuleb esimesena eelrenderdada kaudvarju. Kui sa ei eelrenderda kaudvarju esimesena, võid saada veateate **"No Image to Bake To"** ja sa ei saa midagi selle võre jaoks eelrenderdada ja lisaks pead Blenderi uuesti käivitama.

Sissejuhatus

Tuleb ette olukordi, kus on vaja renderdamiskiirust suurendada, juhtida Blenderit teisest arvutist või kasutada käsurea skripte.

Blenderit saab käivitada ja juhtida ka käsurealt. Sellisel juhul pole tarvis graafilist kasutajaliidest (X serverit Linuxi puhul) ja seda tegevust saab teha ka läbi teise arvuti üle SSH või telneti.

NB! Käsurea võtmed rakenduvad sisestamise järjekorras!

```
blender -b file.blend -a -x 1 -o //render
```

...ei toimi, kuna väljund ja failiformaat on antud enne renderdamise käsku.

-f ja **-a** olgu alati viimased käsurea võtmed.

Süntaks

```
blender [-b <kaust><fail> [-o <kaust><fail>] [-F <formaad>]
[-x [0|1]] [-t <lõimede arv>] [-S <failinimi>] [-f <kaader>]
[-s <kaader> -e <kaader> -a]] [[-P <skripti nimi> [-- <parameeter>]]
```

Renderdamise valikud:

```
-b <kaust><fail>
  Renderdab <faili>, mis on <kaustas> ilma graafilist kasutajaliidest laadimata.
-P <filespec>
  Käivita näidatud Pythoni skript (failinimi või Blenderi sees olev tekst)
-S <nimi>
  Määra stseeniks <nimi>

-f <kaader(frame)>
  Määra kaader, renderda ja salvesta (Ära kasuta võtmega -a)
-j <number>
  Renderda iga x kaadri tagant (jätta vahele näidatud numברי suurune kaadrite hulk)

[-s <kaader>] [-e <kaader>] -a
  Määra alguskaader (-s), lõpukaader (-e) või mõlemad.
  Järjestus on oluline ja on võimalik kasutada ainult -s või -e võtit üksinda.

-o <kaust><fail>
  Määra renderdatud failide salvestamise koht ja failinimi.
  .blend-failiga samasse kausta salvestamiseks kasuta // kui <kaust>.
  Failinimesse kaadri numברי saamiseks kasuta sümbolit #.
  Näiteks: blender -b foobar.blend -o //render_# -F PNG -x 1

-F <formaad>
  Määra renderdamise formaat. Sobivad valikud on:
  TGA IRIS HAMX FTYPE JPEG MOVIE IRIZ RAWTGA
  VIRAW AVIJPEG PNG BMP FRAMESERVER
  (Järgnevad formaadid sõltuvad kompileerimisvalikutest ja platvormist.)
  HDR TIFF EXR MPEG AVICODEC QUICKTIME CINEON DPX

-x [0|1]
  Määrab, kas failinime lõppu lisada ka faililaiend.
  0 tähendab, et mitte, ja 1, et lisada.

-t <lõimede arv>
  Renderdamiseks kasutatavate <lõimede arv>. Reeglina protsessori tuumade arv.
```

Animatsiooni valikud:

(Play-nupu vajutamisel käitub Blender nagu video mängimise programm)

```
-a <valikud> <fail(id)>
  <Faili(de)> mahamängimine toimib ainult siis, kui ei ole kasutatud võtit -b.

-p <sx><sy>
  Ava nii, et alumine vasem serv oleks koordinaatidel <sx>,<sy> (Windowsi all ei toimi)
-m
  Loe kettalt (Ära puhverda)
```

Akna valikud:

```
-w
  Sunni avanema koos aknaraamidega (vaikimisi variant)
-W
  Sunni avanema raamideta (täisekraanirežiimis) (toimib Linuxi/Unixi ja uuema Windowsi puhul)
-p <sx> <sy> <w> <h>
  Ava nii, et vasem alumine nurk oleks ekraanikoordinaatidel <sx>, <sy>
  ning laius ja kõrgus oleksid <w>, <h>
```

Mängumootorispetsiifilised valikud:

```
-g fixedtime
  Joonista ekraanile 50 Hz sagedusega, kaadreid vahele jätmata
-g vertexarrays
  Kasuta renderdamise tippude massiivi (Vertex Arrays) (tavaliselt kiirem)
-g noaudio
  Ilma helita mängumootor
-g nomipmap
  Ilma tekstuuri mipmapita
-g linearmipmap
  Vaikimisi lähima mipmapi asemel lineaarse mipmapi kasutamine
```

Muud valikud:

```
-d
  Lülita silumisrežiim sisse (debug)
-noaudio
  Heli keelamine heli toetavas süsteemis
-h
  Kuvab konsooliaknas sinse abiteksti
-y
  Skriptilinkide mitteväimaldamine. Kasuta -Y, et teada saada, miks tuleb kasutada just -y.
-P <failinimi>
  Käivita määratud Pythoni skript (failinimi või tekst Blenderi tekstiaknas)
-R
  Registreeri .blend-fail edaspidi Blenderiga avanema.
-v
  Kuva Blenderi versioon ja välju.
```

Näited

Pildiks renderdamine

```
# blender -b file.blend -o //file -F JPEG -x 1 -f 1
```

- **-b file.blend**

renderdatav .blend-fail

- **-o //file**

Kaust + pildi väljundfail

- **-F JPEG**

kasuta JPEG-failiformaati

- **-x 1**

lisab failinimele laiendi .jpg

- **-f 1**

Renderda kaader nr 1.

Videoks renderdamine

```
# blender -b file.blend -x 1 -o //file -F MOVIE -s 003 -e 005 -a
```

- **-b file.blend**

renderdatav .blend-fail

- **-x**

Lisab video failinimele faililaiendi .avi

- **-o //images/file**

Kaust + pildi väljundfail

- **-F MOVIE**

Salvestab .avi-faili vähese pakkimisega

- **-s 003 -e 005 -a**

Määra alguskaadriks 003 ja lõpukaadriks 005. ***Tähtis:** võib kasutada -s või -e, kuid vales järjekorras need ei toimi!*

Väljundi sätted

Esimene asi enne renderdamist on väljundformaadi määramine. Väljundformaadi valikute hulka kuuluvad resolutsioon, kaadrisagedus, piksli küljesuhe, failide asukoht ja failitüüp.

Mõõtmed (Dimensions)

Resolution (resolutsioon)

Mõõtmete paneel määrab renderdatud pildi suuruse pikslites.

Vaikimisi on SizeX ja SizeY suurusega 1920×1080 ja neid saab vastavalt X ja Y välja pealt muuta. Nende nuppudega saab muuta pildi suurust.

Protsendi (Percentage) liuguri abil saab renderdatava pildi suurust pikslites protsentuaalselt vähendada. See on sobilik lõpptulemusega sama küljesuhtega testrenderduste tarbeks.

Aspect Ratio (piksli külgede suhe)

Selle all asuvate AspX ja AspY abil saab muuta piksli külgede pikkust vastaval teljel. Vaikimisi on seal arvuti ekraani jaoks sobilik küljesuhe 1:1. Kui toodate sisu teleriekraanil vaatamiseks, tuleb piksli küljesuhet muuta nii, et see klapiks TV-standarditega: PAL Euroopas ja NTSC USA-s.

Piksli küljesuhte kohta vaata täpsemat peatükist [Videoväljund](#).

Border (piiramine)

Terve pildi renderdamise asemel on võimalik renderdada ainult osa kaadrist. Kaameraavaates saad määrata riskülikukujulise renderdatava ala (Border), kui vajutad AltB ja siis lohistad hiirega üle soovitud renderdatava ala.

Tuleb märkida, et sellisel juhul ei saa kasutada järgmiste paneelide funktsionaalsust: Save Buffers, Performance, Full Sample ja Anti-Aliasing.

Linnukese tegemine kärpimisvaliku (Crop) ees lõikab märgitud alast väljaspool oleva ära, nii et musta piirkonda ümber märgitud ala ei jää.

Frame Range (kaadrivahemik)

[Animatsiooni](#) jaoks alguskaadri Start ja lõpukaadri End määramine. Step (samm) abil saab määrata, mitu kaadrit pärast iga kaadri renderdamist edasi liikuda.

Frame Rate (kaadrisagedus)

[Animatsiooni](#) puhul tuleb määrata kaadrisagedus. Vaikimisi on see 24, mis on animatsiooni standard. USA telesüsteemi jaoks tuleb kasutada 29,97 kaadrit sekundis.

Time Remapping (ajanihe)

Sellega saab animatsiooni renderdatavat osa edasi-tagasi nihutada. Kui renderdad uut sama pikka juppi mõnest teisest kohast, saab algus- ja lõpukaadri määramise asemel hakkama ühe numbriga muutmisega.

Formaadid

Töö lihtsustamiseks on sisse ehitatud kogum kõige enam kasutatavaid eelseadistusi. (par = *Pixel Aspect Ratio* - piksli küljesuhe). + ja - nuppudega saab lisada enda formaate:


DVCPRO HD 1080p	1280x1080, 3:2par 24fps
DVCPRO HD 720p	960x720 4:3par 24fps
HDTV 1080p	1920×1080 ruutpiksel 24fps
HDTV 720p	1280x720 ruutpiksel 24fps
HDV 1080p	1440x1080 4:3par 23.98fps
HDV NTSC 1080p	1440x1080 4:3par 29.97fps
HDV PAL 1080p	1440x1080 4:3par 25fps
TV NTSC 16:9	720x480 4:3.3par 29.97fps
NTSC 4:3	720×480 10:11par. 29.97fps
PAL 16:9	720x576 16:11par 25fps
PAL 4:3	720x576 12:11par 25fps

Lisaks formaatides toodud resolutsioonidele saad kasutada ka enda määratud. Piiranguks on arvuti mälu. Eriti suurte piltide renderdamise jaoks vaata üle renderdamisega seonduvad leheküljed.

Väljundi paneel Output

Sellelt paneelilt saad määrata renderdamisel tekkivate failide kvaliteedi ja salvestumise asukoha.

Failide asukohad

Vaikimisi lähevad kõik animatsiooniks renderdatud failid /tmp-kataloogi. Selle muutmiseks tee hiirega nimeväljal ⇧ Shift LMB  ja kirjuta sinna uus nimi. Kui kirjutad sinna // ja uut .blend-faili ei salvesta, pannakse failid Blenderi paigalduskausta.

Kausta ikoonil klõpsimine avab Blenderi failisirvi. Selle abil on mugav kõvakettalt kaustu ja faile otsida ja sirvida.

Raja määramine

Raja määramisel võib kasutada nii tähtedena külge ühendatud kettaid (nt "F:") kui ka suhtelisi failiradu (näiteks "/" ja "/" ja "/" (faili asukoht). Nii kaldkriipsud (/ - Unixi stiilis) kui ka langjooned (\ - Windowsi stiilis) on mõlemas süsteemis vastuvõetavad.

Kirjutamisõiguse olemasolul salvestatakse failid kataloogi, kuhu Blender paigaldati.

Faili tüüp

Blender toetab paljusid pildiformaate. Osadesse failiformaatidesse salvestamiseks kasutatakse Blenderisse sisse ehitatud koodi. Mõnda muusse formaati salvestamisel (näiteks Tiff) kasutatakse Blenderi paigalduskaustas olevaid dll-faile (praeguse näite puhul libtiff.dll). Tähestikulises järjekorras on need (**rasvaselt** on videoformaadid):

Animatsiooniks vajaliku väljundformaadi saab valida Format-paneelilt. Sealt saad valida mitmeid pildi- ja animatsiooni formaate. Pildifailiks salvestamise puhul saad failitüübi määrata ka pärast renderdamist ja enne salvestamist.

Erinevateks puhkudeks on saada mitmeid erinevaid formaate. Pildiformaat võib olla kadudeta või kadudega. Kadudega formaadid kaotavad küll mõningal määral kvaliteedis, aga samas kulutavad need vähem kettaruumi. Kadudeta formaat säilitab iga piksli täpselt sellisena, nagu see alguses oli. Formaadid võib jagada staatilisteks piltideks ja videoklippideks.

Mõlemas kategoorias on nii kommertsettevõtete omandis olevaid formaate ning ka avatud standardeid, mis on mõne ühingu või kogukonna poolt kokku lepitud. Avatud standardid elavad reeglina konkreetsest firmast kauem ja on alati lisatasuta nii programmeerijale kui ka vaatajale. Patenditud formaadid võivad olla piiratud, nii et neid saab töödelda ainult mõne konkreetse videokaardiga. Võib olla ka nii, et koodek on tasuta, aga esitusprogramm maksab raha.

Pakkimine

Mõned formaadid pakivad kettaruumi säästmiseks info kokku. See pakkimine võib olla kadudeta (PNG vms) või kadudega (Jpeg vms). Kadudega formaadid ei säilita infot iga piksli kohta eraldi ja nii kvaliteet väheneb. Formaadid on üsnagi sarnased, kuid igal on oma head ja vead. Sõltuvalt valitud väljundformaadist on paneelil Output näha pakkimise seaded. Näiteks Jpeg-i puhul saad määrata tihendusastet (vaikimisi *Quality*: 90). Kõrgema kvaliteediga pilt kulutab rohkem kettaruumi, kuid on samas vähem ilmsete pakkimise kõrvalmõjudega.

Vaikeformaat on PNG, kuid kuna pilti hoitakse alguses puhvris, saab salvestamisel muu formaadi määrata. (**Tähelepanu**: see kehtib ainult staatiliseks pildiks, mitte animatsiooni renderdamise puhul!)

Kanalid

Blenderiga saab renderdada värvilisi (RGB) pilte, kuid saadaval on ka mustvalge (BW) ja värviline alfa-kanaliga pilt (RGBA). Ära unusta, et kui faililaiendi lisamise nupp pole sisse lülitatud, siis Blender seda automaatselt ei lisa. Nii et näiteks .tga või .png tuleb ise faili salvestamise aknas käsitsi juurde kirjutada.

RGBA valikul tähistab läbipaistvat osa väärtus 0. Läbipaistvuskanaliga faili salvestamisel veendu, et vastav failiformaat seda ka toetab. Näiteks JPEG-formaati salvestamisel pole vahet, kas on valitud RGB või RGBA, sest see failiformaat toetab ainult RGB-d.

Ainsad Z-sügavust tunnistavad failiformaadid on **OpenEXR** ja **Multilayer**. **Multilayer** on ainus formaat, mis toetab renderduskihtide talletamist ja salvestab renderduskäigud eraldi kanalites, nii et neid saab hiljem komposiitoris kasutada.

Blender loeb ja salvestab (Amiga) IFF, Targa, (SGI) Iris ja CDinteractive (CDi) RLE *colormap* formaate. *Colormap*-failitüübi saad valida failitüübi väljalt.

Pildifaili vormingud

BMP	Kadudeta <i>bitmap</i> -formaati vanade joonistusprogrammide päevilt.
Iris	Standardne Silicon Graphics Inc (SGI) formaat, mida kasutati Unix OS-i arvutites.
PNG	Portable Network Graphics, standard, mis loodi GIF-i asendamiseks, kuna see on kadudeta ja võimaldab rohkemate värvide kasutamist. Lisaks toetab see läbipaistvust edastavat alfa-kanalit.
Jpeg	Joint Picture Expert Group (konsortsium, kes selle standardi defineeris). See on avatud formaat, mis tihendab faili väga väikeseks ja sealjuures väga väikeste kvaliteedikadudega. Saab salvestada vaid RGB-väärtusi. Faili mitmekordsel salvestamisel tekib kergesti nähtav kvaliteedikadu.
Jpeg 2000	Kasutab Jpeg 2000 koodekit.
TARGA ja Targa raw	<i>Truevision Advanced Raster Graphics Adapter</i> on lihtne rastergraafikaformaat, mis loodi aastal 1984 ja mida kasutati alguses IBM-i personaalarvutite peal. Toetab läbipaistvuskanalit.
Cineon	Läbipaistvuskanalil salvestamiseks vajuta RGBA-nuppu.
DPX	Sellesse formaati salvestas Kodak Cineon kaamera ja seda kasutatakse kõrgetasemelistes pilditöötlusprogrammides, kõige enam kasutatakse seda digifilmi puhul.
MultiLayer	Digital Moving-Picture eXchange format; avatud professionaalne formaat (lähedane Cineoni formaadile), mis sisaldab ka pildi metainfot; 16-bitine pakkimata bitmap (meeletult suur). Kasutatakse arhiveerimiseks-säilitamiseks.
OpenEXR	OpenEXR-i formaat, mis toetab mitme kihi salvestamist ühte faili. Iga kiht sisaldab üht renderdusastet, nagu näiteks varju, läiget, värvi jne. MultiLayer-formaati salvestamisel saad valida koodeki. Vaikepakkumiseks on kadudeta ZIP. Avatud ja vaba laiendatud ja väga suure dünaamikaulatusega (HDR) pildiformaat, mis toetab läbipaistvust ja Z-sügavust.
Radiance HDR	<ul style="list-style-type: none"> Vajuta <i>Half</i>-nuppu. Vastasel korral salvestatakse värvisügavus 32-bitisena Z-sügavuse (kaugus kaamerast) salvestamiseks vajuta <i>Zbuf</i>-nuppu Kettaruumi säästmiseks vali "koodek". Vaikimisi on selleks ZIP. Läbipaistvuskanalil salvestamiseks vajuta RGBA-nuppu. Kuna OpenEXR on veel nii uus formaat ja operatsioonisüsteemid veel selle eelvaadet teha ei oska, vali Preview (eelvaade), et koos failiga salvestuks ka JPG-formaadis eelvaade.

TIFF Seda kasutatakse sageli faksi piltide puhul

Videomonteerijaga renderdamine

Renderdamine pildijadaks

Paljudel juhtudel võib kodeeritud (encoded) videoriba lõikamine ja ümbertõstmine (montaaž) tekitada tõsist peavalu, sest kodeerimisalgoritm, mida kasutatakse iga pildi taasloomiseks, läheb ühe või kahe või kolme kaadri jooksul 'paigast'. 'Toorete' kaadritega otse töötamine on väga levinud tehnika, mille puhul sa impordid oma video ja renderdad selle üksikute kaadrite jadana, milles iga kaader on salvestatud eraldi pildifaili (enamasti JPG-sse).

Selle saavutamiseks lae menüü *Add->Movie* (lisa->film) kaudu oma algne video. Määra oma formaadi laius (*SizeX*) ja kõrgus (*SizeY*) - kas siis originaaliga samaks, või kui soovid videot moonutada/suurendada/vähendada, siis erinevaks: määra pildi tüübiks JPEG, pane paika kvaliteedi seaded ning määra animatsiooni paneelis (*Anim*) oleva välja *End* (lõpp) väärtus vastavaks videoribas olevate kaadrite arvule. Vajuta nupule *ANIMATION* (animatsioon) ning väljundi paneeli (*Output*) ülemisse failivälja ilmuvad erinevad numbrid.

Võid nüüd videoriba kustutada ning lisada menüü *Add->Image* (lisa->pilt) abil saadud pildid (vajuta parema hiireklahviga kataloogi nimele ja sisse loetakse järjekorras kõik selles kataloogis olevad pildid). Kui sa nüüd lõikad klippi näiteks kaadrist 4321, algab teine riba *päriselt* kaadriga 4322.

Renderdamine videoks

See on absurdsest lihtne (kui õpid selgeks, kus nupud asuvad):

1. Lisa piltide jada, nagu ülevalpool õpetatud
2. Määra oma väljundfaili asukoht ja nimi vastavalt sellele, kuhu soovid seda salvestada (näiteks C:\My Documents\MyMovie), renderdamisnuppude all olevas ülemises väljundikastis.
3. Muuda oma formaat videofaili formaadiks (AVI, MOV, FFMPEG) ja määra kodeerija (*Codec*)
4. Määra kaadrisagedus vastavalt selle, millise sagedusega sinu jada tuleks mängida. See asub animatsiooni/mahamängimise nuppude juures.
5. Määra animatsiooni lõpp (*End*) vastavalt sellele, kui palju pilte jadas on.
6. Animeeri (*ANIM*)

Luuakse ja salvestatakse üks video; selle nimi vastab sinu poolt valitule, kuid selle lõppu on lisatud algus- ja lõpukaadri numbrid (näiteks MinuFilm0000-0250.avi)

Video ettevalmistamine

Pärast animeerimise selgekssaamist tahad sa kindlasti teha ägedaid animatsioone, salvestada need oma lemmikkoodekiga videoks ja arvatavasti ka näidata oma loomingut läbi interneti.

Varem või hiljem tekib vajadus teha animatsioon televisiooni või ise DVD kõrvetada. Tulevaste probleemide vältimiseks on siin mõned nipid video ettevalmistamiseks. Esimene ja kõige tähtsam on märgata topeltkatkendjoonitud ala kaameravaates!

Kui renderdad arvutis vaatamise jaoks, on nähtav kogu renderdatav pilt koos väljaspool katkendjoont olevaga. Vanadel kineskoopitel on omadus, et osa pildist jääb kineskoobi elektronkiire iseärasuste tõttu "serva taha". Kui sa jätad vajalikud osad katkendjoonest sissepoole, võid kindel olla, et need on ka kineskoopekraanil nähtavad. Tegeliku serva ja katkendjoone vahel olev ala võib olla nähtav või nähtamatu. See sõltub vaatajate teleritüübist ja seadistustest.

Renderdatav ala on rangelt TV-standardi poolt paika pandud. Blenderis on mitu põhilist eelseadistatud formaati, muu hulgas:

- PAL 720×576 pikslit küljesuhtega 54:51.
- NTSC 720×480 pikslit küljesuhtega 10:11.
- PAL 16:9 720×576 küljesuhtega 64:45, mis on mõeldud 16:9 küljesuhtega telerite jaoks.
- HD 1920×1080 pikslit küljesuhtega 1:1 ja veel mitu HD-video standardit.

Kui kasutad lõpptulemust DVD-l, ei ole mõtet renderdada näiteks 1600×1200 pikslilist pilti. DVD-formaat ei toeta nii kõrget lahutust ja skaleerib pildi väiksemaks ja standardiga sobivaks. Lõpptulemusest suuremalt renderdades raiskad praegusel juhul umbes 4 korda rohkem kettaruumi ja renderdusele kulub vajalikust 4 korda rohkem aega.

Pikslil külgede suhe

Vanadel kineskoopitel pole pikslid ruudukujulised, vaid pisut lapikud või kitsamad, sõltuvalt TV-standardist. Nii et nende jaoks tuleb valmistada "eelsurutud" pilt, mis näeb hea välja teleekraanil, aga mõnevõrra kehvem arvuti ekraanil. Hea pildi saamiseks on oluline kasutada õiget küljesuhet, et pilti hiljem kokku ei surutaks ega välja venitataks.

Värviküllastus (*Saturation*)

Enamus videokassette ja videosignaalidest kasutab RGB asemel YCrCb mudelit: täpsemalt YUV Euroopas (PAL) ja YIQ USA-s (NTSC), mis on eelnevale küllalt sarnane. Siinkohal mõningane vajalik teave.

YCrCb mudel saadab informatsiooni hele-tumedusena ehk valgustugevusena ja kahe värvikomponendina, milleks on punane ja sinine (Cr ja Cb). Mustvalge teler näitab ainult hele-tumeduse informatsiooni (*Luminance*). Värviteler aga lisab heletumedusele veel värvitooni informatsiooni (*Chrominance*). YCrCb väärtuste RGB-ks teisendamine toimub kahes osas (*kaldkirjas* kirjutatu sõltub standardist: PAL või NTSC):

Esiteks Gamma korrigeerimine (*g* varieerub: 2.2 NTSC-s, 2.8 PAL puhul):

- $R' = R^{1/g} \cdot G' = G^{1/g}$
- $B' = B^{1/g}$

Seejärel konverteerimine ise:

- $Y = 0.299R' + 0.587G' + 0.114B'$
- $Cr = a_1(R' - Y) + b_1(B' - Y)$
- $Cb = a_2(R' - Y) + b_2(B' - Y)$

24-bitise RGB pildi puhul kasutakse iga kanali jaoks 8 bitti. Kuid kuna inimsilm on tundlikum hele-tumedusele ja vähemtundlik värvitoonile, kasutatakse hele-tumeduse edastamiseks rohkem bitte. Tulemuseks on see, et videos on värvide dünaamika väiksem, kui võimaldaksid arvuti monitorid. Jäta meelde, et absoluutselt kõiki värve ei ole võimalik õigetena kuvada.

Põhireeglina tuleb jätta värvid nii halliks või värvituks kui vähegi võimalik. Nii saab umbes 80% ulatuses konverteerimisel dünaamika samaks jätta. Teisisõnu ei tohiks ükski RGB väärtus ületada arvu 0,8 ([0–1] skaalal) või ei tohiks ületada väärtust 200 ([0–255] skaalal).

See ei ole üks-üheselt võetav. Mõnikord on suurem kui 0,8 vastuvõetav, kuid RGB ekraanil, mis on seadistatud kontrastivahemikule 0,0 kuni 1,0, näeb see välja kole (liiga kontrastsed värvid). Samas võib see arvuti ekraanil välja näha särav ja dünaamiline.

Poolkaadritena renderdamine

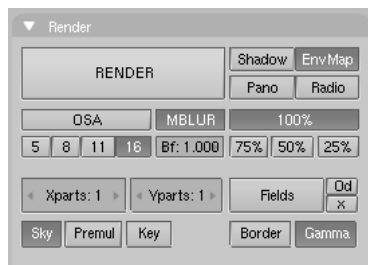
TV-standardis on kirjas, et PAL-i puhul see 25 kaadrit sekundis ja NTSC puhul 30. Kuna kineskoopekraanis olev fosfor ei helenda väga kaua, oleks sellisel ekraanil pilt väga värelev.

Väreluse vähendamiseks mõeldi välja selline kuvamise meetod, et kuvatakse üle ühe rea. Nii saab kahekümne viiest sekundis 50 poolkaadrit sekundis ja NTSC-s kolmekümnest 60 poolkaadrit sekundis. Selline kaadrisageduse valik on tingitud sellest, et juba elektrifitseerimise algusaegadest on Euroopas võrgusagedus 50 Hz ja USA-s 60 Hz.

"Interlaced" tähendab sisuliselt seda, et üks "poolkaader" tähendab kõiki paaris ja teine "poolkaader" paarituid ridu.

Kuna poolkaadritel on märgatav ajavahe, ei saa neid renderdada nii, et teha valmis üks täiskaader ja see siis kaheks ära jagada. Nii tehes oleks liikuvate objektide servad vägagi märgatavalt "tribulised".

Valikud



Poolkaadrite seadistamine.

- Fields (poolkaadrid)

Lülitab sisse poolkaadritena renderdamise. Kui nupp Fields on paneelil Render sisse lülitatud, renderdab Blender iga kaadri kahe käiguna. Kõigepealt renderdatakse paaritud read, liigutakse ajas poole kaadri võrra edasi ja renderdatakse paaris read. Tulemuseks on kehv "triibuline" pilt arvuti ekraanil ("Poolkaadrite kaupa renderdamise tulemus"),

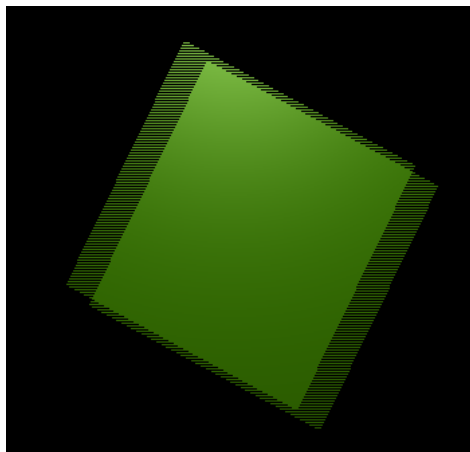
aga kineskoopiteleri ekraanil näeb see välja väga hea.

- Upper First / Lower First

Siit saab määrata, kas esimesena renderdatakse ülemised või alumised read.

- Still

Lülitab välja poolkaadrite renderdamisel ajavahega arvestamise (x).



Poolkaadrite kaupa renderdamise tulemus.

Korrektne ridade järjestus

Blender renderdab vaikimisi esmalt paaris ja seejärel paaritud read. Nii on see määratud PAL-standardis. NTSC-süsteemis alustatakse paaritutest.

Kui sa valid siin vale järjekorra, on tulemuseks poole hullem tulemus, kui oleks üldse ilma poolkaadriteta renderdamine!

Kui sa kahtled, kumb on õigem, renderda valmis üks lihtne vasemalt paremale liikuv kuubik mustal taustal. Tee üks versioon paaris ridadega alustades ja teine paaritutega. Õige variant näeb välja sujuv ja vale kohutav. Nii saad vältida "tundide kaupa" raisatud renderdamisaega.

Poolkaadrid ja komposiitsõlmed

Sõlmed (*nodes*) hetkeseisuga poolkaadreid ei tunnista. See on osaliselt tingitud sellest, et poolkaadrite puhul on väga keeruline naaberpikslitega opereerimine (näiteks hägustamise, vektorhägustamise efektide puhul). Lahenduseks on poole tihedama kaadrisagedusega renderdamine ja alles hiljem poolkaadriteks tegemine.

Videoformaadid

Neid formaate kasutatakse põhiliselt lõpptarbija jaoks mõeldud failide tegemiseks.

Koodek on programmijupike, mis teeb videovoo väiksemaks, nii et see mahuks DVD-le või nii et seda saaks otsepidina üle interneti edastada. Koodek surub videokanalid kokku, et säästa kettaruumi, ja püüab tagada katkestusteta esitamist. Kadudega (*lossy*) koodek suudab failid väiksemaks teha, aga seda kvaliteedi arvelt. Mõned koodekid, nagu näiteks H.264, on sobivad suurematele piltidele. Koodekit kasutatakse nii kokkusurumisel kui ka avamise, nii et koodek peab olemas olema nii Blenderiga töötaja arvutis kui ka hiljem videot vaatava inimese arvutis. Kokkusurutud info talletatakse konteinerformaadis.

Blender tunnistab kaht tüüpi konteinerformaate:

- Audio Video Interlace (.avi-laiendiga) ja
- QuickTime (.mov laiendiga).

AVI konteineri valimisel tekib sinna koodeki valimise hüpikaken. Seal kuvatakse selle arvutikasutaja arvutisse installeeritud koodekid. Igal koodekil on unikaalne valik seadistusi. Täpsema info saamiseks pead lugema koodeki tootja poolt antavat informatsiooni.

Quicktime'i valikul ilmuvad valikuks arvutis olevad koodekid ja sa saad valida endale sobiva. Selle valimiseks võib olla tarvilik Quicktime Pro litsentsi ostmine.

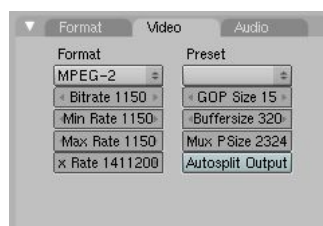
Koodekeid on olemas sadu, nende hulgas XviD, H.264, DivX, Microsoft jne. Igal neist on omad head küljed, halvad küljed ja erinev ühilduvusaste igasugu mängijate ja operatsioonisüsteemidega.

Mõned koodekid toetavad ainult RGB- või YUV-värviruumi, mõned toetavad ka läbipaistvuskanaalit. RGBA-d toetavad koodekid on:

- animation (quicktime)
- PNG *TIFF *Pxllet - kadudega ja võib olla mängitav ainult Apple Maci arvutis.
- [Lagarithi kadudeta videokodek](#)

AVI Codec	AVI koodeki pakkimine. Saadaolevad koodekid sõltuvad operatsioonisüsteemist. Pärast AVI-konteineri valimist kuvatakse koheselt koodeki valimise aken. Koodekit saab muuta nupuga Set Codec (vali koodek).
AVI Jpeg	JPEG-kompressiooniga AVI-fail. Kadudega väike fail, mis pole nii väike, nagu saaks koodekit kasutades. JPEG-kompressiooni kasutatakse ka DV-videot salvestavates kaamerates.
AVI Raw	<i>Audio-Video Interlaced</i> (AVI) pakkimata kaadrid.
Frameserver	Blender väljastab tellimusel kaadreid osana renderdamisfarmist . Pordi number on määratud OpenGL-i all paneelil Preferences.
H.264	Pakib videod H.264 koodekiga. Vaata pealkirja Pakkimine edasijõudnutele alt.
MPEG	Pakib videod MPEG koodekiga. Vaata pealkirja Pakkimine edasijõudnutele alt.
Ogg Theora	Pakib videod Theora koodekiga Ogg-failideks. Vaata pealkirja Pakkimine edasijõudnutele alt.
QuickTime	Apple'i Quicktime'i .mov-fail. Quicktime'i koodeki valimise dialoogiaken avaneb kohe pärast Quicktime'i konteinerformaadi valimist. Vaata Quicktime'i pakkimine
	Loeb GIF-faili siis, kui Quicktime on installeeritud
	Blender suudab GIF-faili avada Windowsi ja Maci platvormil, siis kui [QuickTime] on paigaldatud. Quicktime pakub võimaluse lugeda GIF-faile (nagu ka kihtideta PSD-faile).
Xvid	Pakib videod Xvid-koodekiga. Vaata pealkirja Pakkimine edasijõudnutele alt.

Pakkimine edasijõudnutele



Kui on valitud H.264, MPEG, Ogg Theora või Xvid koodek, ilmub nähtavpakkimise (Encoding) paneel. See lubab muuta pakkimise seadistusi neile formaatidele, mis toimivad läbi FFMPEG.

FFMPEG, lühend sõnadest *Fast Forward Moving Pictures Expert Group*, on kogum avatud lähtekoodiga tarkvarajuppidest, mis suudavad eri formaatides videot salvestada, konverteerida ja voona edastada. See sisaldab programmijuppi libavcodec, mida kasutatakse ka paljudes teistes projektides. Lisaks sisaldab see programmi libavformat, mis toimetab heli/video konteinerformaatide multipleksimist ja demultipleksimist.

Video seadistused (Video Settings)

Siit saab valida kasutatava koodeki ja tihendusseaded. Kõigi nende valikute juures varieeruvad failisuurus, ühilduvus erinevate platvormidega ja esitamiskvaliteet.

Võid kasutada eelseadistusi, DV, SVCD, DVD ja teiste jaoks. Seal on paika pandud optimaalsed seadistused vastavale formaadile (MPEG-1, MPEG-2, MPEG-4, AVI, Quicktime (kui on installeeritud), DV, H.264 või Xvid (kui on installeeritud)). Vastava koodeki kasutamiseks peab see Blenderiga samasse arvutisse installeeritud olema. Siis saab Blender seda koodekit kasutada.

Videoformaadid:

MPEG-1	...
MPEG-2	...
MPEG-4	...
AVI	...
Quicktime	...
DV	...
H.264	...
Xvid	...
Ogg	...
Matroska	...
Flash	...
Wav	...
Mp3	...

Videokodekid:

Puudub	...
MPEG-1	...
MPEG-2	...
MPEG-4(DivX)	...
HuffyUV	...
DV	...
H.264	...
Xvid	...
Theora	...
Flash Video	...
FFmpeg videokodek #1	...

Bitrate (bitikiirus)

..

Rate (kiirus)

...

Minimum (miinimum)

...

Maximum (maksimum)

...

Buffer (puhver)

...

GOP Size (GOP suurus)

...

Autosplit Output (automaatne jupitamine)

Kui videofail on nii suur, et ületab kahe gigabaidi piiri, lülita sisse automaatne jupitamine (Autosplit Output). Põhiline meetod failisuuruse muutmiseks on GOP-suuruse muutmine. Ehk siis see, kui suur on pakkimisel eraldamatute kaadrite grupp. Suurem number loob tavaliselt pisema faili, aga nõuab ka arvutit mahamängimiseks suuremat ressursi.

Mux (multipleksimine)

...

Rate (kiirus)

...

Packet Size (paki suurus)

...

Standardid

Videokodekid ei tunnista seinasuuruseid kaadreid: jälgi võimalusel X ja Y suuruse määramisel TV-standardeid.

Heli seadistused (Audio Settings)

Heli pakitakse vastavalt valitud kodekile.

Helikodekid

MP2	...
MP3	...
AC3	...
AAC	...
Vorbis	...
FLAC	...
PCM	...

Bitrate (bitikiirus)

Iga kodeki puhul on võimalik määrata, kui suurt andmevoogu see kasutab ja kui kvaliteetne seega heli on. Näidis kasutab MP3 pakkimist andmevoogu suurusel 128 kbps. Suuremat andmevoogu on kehvem reaalajas voona edastada, kuid selle kvaliteet on parem. Ühilduvuse huvides kasuta suurendamisel 2 astmeid

Samplerate (ximissagedus)

Diskreetimissagedus määrab kasutatavate helisämplite arvu sekundis. Vaikimisi väärtus on 44100, mis on standardne suurus mitmete failitüüpide juures, näiteks CD-plaadil, ja see annab kõrge kvaliteediga heli.

Volume (helitugevus)

Määra väljundi helitugevus.

Nipid

Pildiformaadi valik sõltub sellest, mida pildiga hiljem ette võtta kavatsetakse.

Seisvate piltide korral

- sõpradele e-mailimiseks on sobilik JPG
- kui on plaanis seda pilti veel järeltöötluse käigus taustaga kombineerida ja tarvilik on ka läbipaistvuskanal, vali PNG
- kui on tarvis veel komposiitoris sügavusteravust ja udustamist kasutada, sobib **EXR**
- renderduskäikudega komposiitimiseks, näiteks *Vector pass*, sobib **Multilayer**.

Kui järeltöötlust ja eriefekte vaja pole, vali konteinerformaadiks AVI ja koodekiks AVI-JPEG või avatud Xvid-koodek. Kui tahad videot koos heliga veel järjestajas monteerida, vali **FFMPEG**.

Järeltöötluse tegemise korral on parim variant renderdamiseks kaadrite kaupa PNG-pildijadaks salvestamine; kui aga soovid kaadreid saada ühe failina, vali **AVI Raw**. Kuigi toores Raw AVI on hiiglaslik, säilitab see järeltöötluse jaoks tarviliku algse kvaliteedi. Alles pärast järeltöötlust (komposiitoris ja järjestajas) tasub kasutada tihendatud failiformaati. Järeltöötluseks ei sobi tihendatud failiformaat, sest pakkimise kõrvalmõjud võivad soovitud efekti saamise võimatuks muuta.

Jäta meelde, et animatsiooni videoks (AVI või Quicktime) renderdamise asemel on parem renderdada piltide jada ja siis need pildid videoks kokku liita. Probleemi ilmnemisel tuleb otse videoks renderdamisel kogu renderdust otsast pihta alata. Pildijadaks renderdamisel saab aga katkestatud koha pealt jätkata.

Efektid järeltöötlusega

Renderusvalikute alt saad valida mitmesuguseid elemente, mis lisatakse pildile pärast renderdamise lõppu. "Kaamerast tulnud" pildidel neid ei ole, need lisatakse hiljem komposiitoriga.

Valikute Compositing ja Sequence kohta on kirjutatud peatükis [Väljundi valikud](#).

Poolkaadreid (Fields) kirjeldatakse peatükis [Videoväljund](#).

Renderduskihid (*Render Layers*)

Renderduskihtide abil saab kokkuliidetud pildi kihtideks jagada. Renderduskihte tuleb kasutada mingil kindlal eesmärgil, nagu näiteks sügavusteravuse kasutamine, mingite kindlate osade teistmoodi valgustamine, tonaalsuse muutmine mingis pildi osas jne. Märksõnaks on eraldamine. Renderduskihid võimaldavad üksikuid elemente isoleerida: näiteks selleks, et neid ei peaks muude asjade muutmisel iga kord uuesti renderdama, või selleks, et mingi osaga ülejäänud tervikust midagi erinevat ette võtta. See aitab proovirenderduste ajal aega kokku hoida.

Renderduskihid komposiitori kasutamisel

Mille jaoks on renderduskihid mõeldud? Blenderi sõlmedel põhineva komposiitori jaoks!

Kui lisad komposiitoris renderduskihi sõlme (*RenderLayers*) ja valid selle alt vastava renderduskihi, saad mängu tuua ükskõik mis informatsiooni, mis selles renderduskihis olemas oli. Sellest sõlmest saab kogu renderduskonveieri algsõlm (*vaata altpoolt*) ja see on objektidega seotud renderduskihi või kihtidega. Iga vaheetapp liigub siis mööda ehitatud sõlmestikku väljundi suunas.

Kas kihid või käigud (*Layers* või *Passes*)?

Blenderi [renderduskäikude](#) süsteem on renderduskihtide alamhulk. Renderduskäigud on erinevad varjutamise elemendid nagu värv ja läige, mida saab hiljem komposiitimisel kombineerida. Renderduskihid on rohkem stseeni komponentide eraldamisele suunatud; samas saab nendega ka renderduskäike eraldada.

Renderduskihtide kasutamine

Ava renderdusnuppude alt sakk Layers. Siin valid renderdatavad kihid ja tulevaste renderduste seadistused.

Aktiveerimine ja nimetamine

Valikukastis on sinu loodud renderduskihid ja võimalus kihte deaktiveerida, kustutada, lisada ja ümber nimetada.

Olgu märgitud, et saki Layers all olevad seadistused on renderduskihispetsiifilised. Veendu, et seadistuste muutmisel oleks valitud õige renderduskiht.

Märkeruuduke lülitab renderduskihi arvutamise sisse või välja. Aja kokkuhoiu huvides luba ainult kihid, mil parajasti töötad. Sirvija võimaldab olemasolevate renderduskihtide läbisirvimist ja uute lisamist.

Uue renderduskihi tekitamine

Vaikimisi on olemas 1 renderduskiht, mis sisaldab kõiki kihte, olgu need siis stseenis kasutusel või mitte. Uue renderduskihi lisamiseks vajuta plussmärgi kujulist nuppu. Nüüd saad sa valida kahe renderduskihi vahel. Aktiivset kihti näidatakse aknas. Igal renderduskihil on oma valik renderdatavaid kihte (kuidagi loogiline või mis?).

Näiteks võib stseenis olla robot, koos muude maapinnal olevate objektidega nagu majad jne. Kui robot asub nähtaval kihil 5, võid teha renderduskihi nimega "Robot", mille puhul oleks valitud kiht 5 nii stseenis kui ka renderduskihtides.

Siis võid näiteks teha järgmise renderduskihi nimega "Muu kraam" ja tolle puhul jätta valituks kõik teised kihid peale kihi 5. Seda nii stseenis kui ka renderduskihtides. Edasi võib sõlmedaktoris (Node Editor) kasutada kahte renderduskihi tüüpi (*RenderLayer*) sisendsõlme: ühe sisendiks renderduskiht "Robot" ja teise sisendiks "Muu kraam". Seejärel võib need kaks sõlme liitva sõlme abil tervikpildiks kokku arvutada.

Stseeni kihtide seaded

Stseenikihtide nuppe on kolme sorti:

Scene (stseen)



Need peegeldavad kihinuppe 3D-vaate aknas ja nendega määratakse ära renderdatavad kihid.

Layer (kiht)

Määrab ära, millised stseenikihid on eesolevas renderduskihis.

Mask Layers (katvad kihid)

Renderdatud pilt pärineb objektidest, mis on märgitud kihtide ja Z-maski kihtide vahel. Selles näites on kuubik kihis 2 ja 3, muru on kihis 1. Nüüd meie poolt nimetatud renderduskihis "zmask", nagu näha ülal olevalt pildilt, kiht 1 on valitud ja kiht 3 on määratud kui Z-mask (seda näitab musta täpi olemasolu). Tulemusena renderdatakse ainult see osa kihist 1, mis on eespool kihis 3 olevast objektist (kuubikust).

Selle kihi saab valida vasakpoolse LMB  hiirenupuga. Mitme kihi valimiseks tuleb lisaks all hoida Shift-klahvi ⇧ Shift LMB . (Sel puhul muutub nupul olev mullike *tumehalliks*.)

Kihtide valikud

Renderdatakse ainult need objektid, mille kihid on sisse lülitatud NII stseeni kihi nuppudest kui ka renderduskihi kihtide nuppudest. Seega: kui stseenis on ainult kiht 1 valitud ja sinu renderduskihis on määratud renderdatavateks ainult kihid 2 ja 3, siis peale taeva midagi ei renderdata (eeldame, et oled taeva teinud).

Ümberlülitused (*Overrides*)

Valgustuse ja materjali valiku kastide abil saab teha nii, et kõik objektid sellel kihil oleks valgustatud ainult neis määratud valgustigrupiga või et kõik objektid sellel renderduskihil oleksid seal määratud materjaliga.

Light (valgus)

Sisesta siia valgustigrupi nimi ja pärast seda valgustatakse kogu stseeni ainult tolles grupis olevate valgustitega. Tavaliselt kasutatakse seda proovirenderduste kiiremaks kättesaamiseks. Kui stseeni valgustus on väga keeruline, saab esialgse proovirenderduse siitkaudu paari põhivalgustiga ära teha.

Material (materjal)

Siia saab panna materjali, mida siis kasutatakse kõigi stseenil olevate objektide puhul. See on jällegi hea proovirenderduste jaoks. Esiialgse valgustuse katsetamiseks kasuta vaikematerjali.

Sisalduvad valikud (*Include Options*)

Iga renderduskihi juures saab valida, milliseid omadusi renderdatakse. Aja kokkuhoiuks ja renderdusastmete määramiseks saab nende nuppude abil põhiomadusi sisse ja välja lülitada.

Z-mask

Renderdab ainult kehade Z-väärtusest eespool oleva.
Negate

Renderdab ainult kehade z-väärtusest tagapool oleva.

AllZ

Z-väärtus arvutatakse kõigele nähtavale, mitte ainult renderdatavatele. Kui see on välja lülitatud, siis kihti mittekuuluvatel objektidel z-väärtus puudub (nad on "lõpmatuses").

Solid (keha)

Renderdatakse küljed (*face*). Kõik tavalised võred koosnevad materjaliga külgedest.

Halo

Renderdatakse halo materjal

Z-transp (Z-läbipaistvus)

Läbipaistvus võib olla Z-väärtusel põhinev või kiirtejälitajalt saadud. Kui see on Z-väärtusel põhinev ja sisse on lülitatud "Ztra", renderdatakse läbipaistvad alad läbipaistvast alast tagapool oleva z-väärtusega.

Sky (taevas)

See renderdab taeva vastavalt materjali maailma seadistustele. Vastasel korral renderdatakse must läbipaistev taust

Edge (serv)

Selle valikul renderdatakse renderduskihi objektidele servajoon. Selle sisselülitamisel võetakse servajoon tekitamise seadistused Render saki Output sektsioonist. Servajoonde tegemine tuleb sakist Output samuti sisse lülitada.

Strand (karvad)

Karvad on staatilised osakesed, mida toonitakse materjali seadistuste kohaselt; need näevad välja kui juuksesalgud või muru.

Renderduskäigud (*Passes*)

Renderduskäikudest (*Combined*, *Z*, *Vec* jne) on juttu [järgmisel lehel](#).

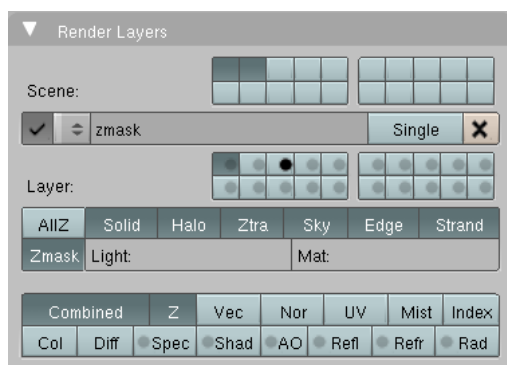
Näited

Ainult mõningate objektide renderdamine

Oletame, et sa oled teinud roboti, lisanud sellele halo ja sa tahad ruttu näha, kuidas see välja näeb. Oletame, et su stseenis on kihil 1 kastid, lasermõõgad kihil 2, robot kihil 5 ja valgustid ning kaamerad asuvad kihil 20. Kõik need on 3D-vaates nähtavad. Kui tahad ainult robotit renderdada, klõpsad kiht 5 nuppu (mis on renderduskihi nime all), muudad taeva mitterärgituks (nii et taevast ja horisonti ei renderdata) ja märgid ära halo. Valma! Renderdamisel renderdatakse (kiirelt) ainult robot ja kõik muud asjad jäetakse välja (nagu näiteks kastid, mille ees robot jooksis).

Ainult valitud objektidele äärejoone tegemine

Ainult mõningatele objektidele äärejoone renderdamiseks liiguta need kõigist teistest eraldi kihile. Tekita 1 renderduskiht, kus on ainult vajalikud kihid. Tee ülejäänud objektidele teine renderduskiht. Esimesel kihil lülita Edge sisse (pea meeles, et see tuleb sisse lülitada ka Output sakil) ja veendu, et serva renderdamine oleks teise renderduskihi jaoks välja lülitatud. Sõlmeredaktoris tee kaks sisendsõlme, üks kummalegi renderduskihile. Miksi need kaks kokku. Tehtud. Lihtne. Juhõissa!



Renderduskäigud on kõikvõimalikud renderdaja poolt pakutavad väljundid. Kõik eraldatud renderdused ühendatakse lõpuks tulemuseks ja seda nimetatakse programmis ühendatud (*Combined*) väljundiks. Kuid iga renderduskäiku võib ka iseseisvalt väljundiks kasutada. (Enamikul juhtudest, kui seda teed, saad valida, kas seda lisaks ka ühendatud väljundis kasutada.)

Et näha tulemusi, peavad mõned neist väljunditest olema sisse lülitatud ja ka stseenis kasutusel (ning mitte ainult renderuskihi paneelilt valitud olema). Näiteks kui stseenil pole ühtegi valgustit või need valgustid on määratud varje mitte tegema või objektide materjalid on määratud valguskiiri ignoreerima, on tulemuseks tühjus; pole mitte midagi näidata. Kui kaudvari (*Ambient Occlusion*) on Maailma alt välja lülitatud, on **AO** kõik tühi. Seda isegi siis, kui sa selle valinud oled.

Aja säästmiseks pead renderduskihtide paneelilt renderduskihid ise määrama (renderduskihte tutvustab [eelmine peatükk](#)):

Combined	See renderdab kõik, isegi mittevajaliku. (N-ö "kogu kupatuse".) See oleks siis kõikide valikute valimisel, ühte väljundisse kokku sulatatuna. Ainult kaamerakujulise nupuga väljalülitatud jäetakse välja.
Z	Z-sügavuse kaart; kui kaugel mingi piksel kaamerast asub. Sügavusteravuse kasutamine (Depth-Of-Field - <i>DOF</i>). Sügavuskaart on lineaarne pöördväärtus (<i>1/kaugus</i>) kaamera nägemisulatus algusest (<i>clipping start</i>) alates.
Vector	Pikslite liikumissuund ja kiirus. Kasuta koos vektorhägu (Vector Blur) sõlmega.
Normal	Arvutab välja pinnafaktuuri nähtava geomeetria (pilt, mille abil antakse edasi võltsdetailsust), et muuta pinnale langeva valguse näivat suunda.
UV	Lubab pärast renderdamist tekstuurida. Vaata UV-sõlme alt.
Mist	Edastab udu tekitamise käigu
Object Index	Maskib valitud objektid. Vaata MaskObj sõlme.
Color	Materjali värv ilma varjutuseta
Diffuse	Materjali pinnal hajuv valgus
Specular	Valgusallika peegeldumine
Shadow	Varjude tekitamine. Jälgi, et varjude tekitamine oleks valgusti poolt lubatud (positiivne või negatiivne) ja et materjalid ka varju vastu võtaksid. Selle käigu kasutamiseks korruta see Diffuse-käiguga Mix sõlmes.
Emit	Helendamise/kuma käik
AO	Hajusvalguse käik. Jälgi, et see oleks keskkonna all sisse lülitatud ja et kiirtejälitus (<i>RayTracing</i>) oleks sisse lülitatud.
Environment	Keskkonnavalgustus.
Indirect	Kaudvalguse käik.
Reflection	Peegeldused peeglitelt ja teistelt peegeldavatelt pindadelt (näiteks vahatatud heledad põrandad). Selle kasutamiseks sega see sõlmes Mix käiguga Diffuse, kasutades parameetrit <i>Add</i> (Lisa).
Refraction	Valguse murdumine läbipaistva võre läbimisel. Selle kasutamiseks sega see sõlmes Mix käiguga Diffuse, kasutades parameetrit <i>Add</i> (Lisa).

Käigu võimaldamisel ilmub see maagiliselt uue väljundina sõlme külge ja seda saab kasutada nii nagu näidisel näha.

Renderduskäikude vahelejätmine

Nagu varem öeldud, on *Combined* väljund sulam mitmetest eraldi olevatest asjadest. Kui ühe neist väljunditest valid, antakse nad välja eraldi ning ühtlasi ka kombineeritud tulemuses.

Kui lülitad käigu juures oleva kaamera kujulise nupu sisse, siis seda käiku lõplikku pilti ei panda. Käigud on olemas küll väljundina, aga neid ei kaasata *Combined* käiku.

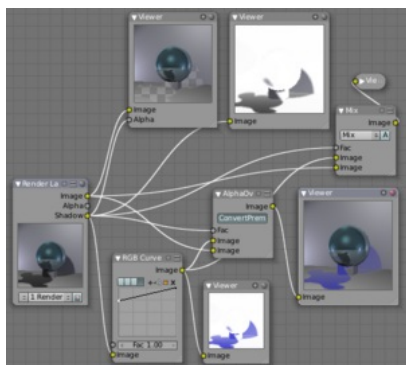
Renderduskäikude kasutamine

Renderduskäikude põhieesmärk on see, et neid saaks sõlmede töötlejas mitmesuguste tulemuste saamiseks võrguks ühendada. Nende kasutamisel on võimalik saada väga erilisi efekte ja samuti saad oluliselt kokku hoida ka renderdamiseks kuluvat aega. Mõne aja pärast vaatame näiteid.

Pisut informatsiooni tavapäraste käikude kasutuse kohta on mujal:

- *Image*: kuna see on põhitoode, kasutatakse seda Blenderis kõikjal.
- *Alpha*: vaata sõlmi *AlphaOver* ja *Matte*
- *Z*: vaata sõlme *Defocus*
- *Vec*: vaata sõlme *Vector Blur*
- *Normal*: vaata sõlme *Normal'*

Varjude toonimine



Varju toonimiseks tuleb varju käik lasta läbi *Colorization* sõlme ja siis see jälle muu pildiga kokku ühendada. Tulemuseks on ebatavaliselt värvilised varjud. Vaatame sammhaaval parempoolsel pildil olevat sõlmestikku. Vasemal on renderduskihi sisendsõlm: see viitab ühele meie poolt stseenis tekitatud renderduskihile. Stseenis on alusel peegeldav pall ja selle taga taustasein. Kõik (peale palli) on hall. Kasutame tavapärast nelja valgustiga valgustusparki: taustavalgustus kõrgel, kaks küljevalgustit aluse kõrgusel ja peavalgusti asub kõrgel üleval kaamerast pisut vasemal. Pärdikukujuline võre Suzanne hõljub peavalgusti ees, heites põrandale varju. Pallilt tekib vari põrandale ja taustaseinale.

Renderduskihi sõlme erinevad väljundid on ära määratud sellega, millised nupud me renderduskihti määratledes valisime. Ülemised kaks vaaturit näitavad selliseid väljundeid nagu vari ja läbipaistvuskanaal. Kohe nende kõrval on varju kanal. Vasemas vaates on varju tumedates osades pilt läbipaistev. Nüüd on varju abil osa pilti välja lõigatud.

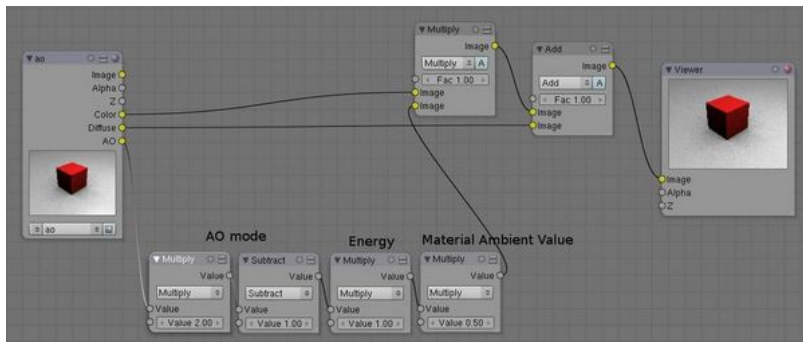
Saadame varju läbi RGB kõvera, mis on seadistatud suurendama ainult sinise hulka 75%, nii et hall vari (R:40, G:40, B:40) muutub (R:40, G:40, B:40×1.75=70). Siniseks värvitud vari on näha allservas olevas vaaturis. Nüüd on meil kaks valikut: AlphaOver ja Mix. Variantid on sellised:

- Kasuta varjukaarti parameetris *Factor*.
- Saada sinine vari ülemisse pessa
- Saada aluspilt alumisse pessa

Tulemus on kõigil juhtudel sama: sinine vari. Pane tähele, et Suzanne'i peegeldused ei ole "õiged"; need pärinevad hoopis teisest renderduskäigust.

Selle pildi võib asendada ka hoopis teisega; näiteks varjukaardiga teiselt renderduskihilt. Teistsuguse efekti saamiseks võid Image Input-sõlme abil sisse tuua pildi hoopis teisest projektist. (See oleks samane "Star Wars Episode One" filmiposterile, kus Anakin Skywalkeri taga on Darth Vaderi vari.)

Kaudvarju (*Ambient Occlusion*) komposiitimine



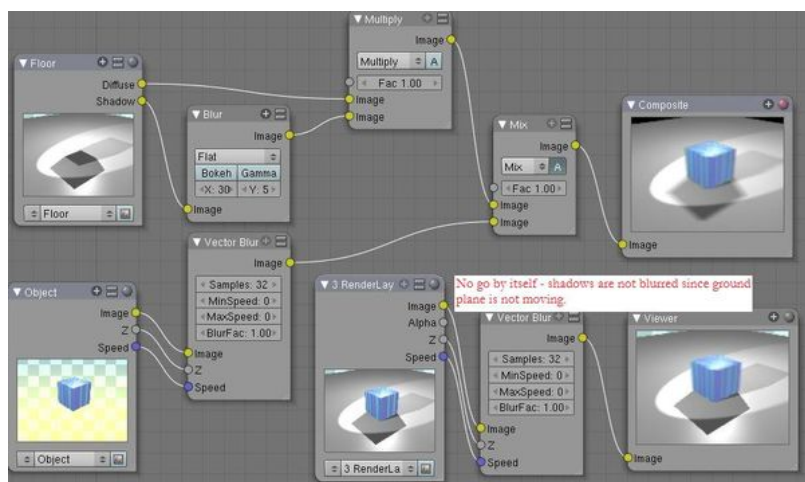
AO on geomeetriapõhine varjutaja, mis teeb sisenurgad tumedamaks. Selle saab sisse lülitada Maaailma seadistuste alt ja see arvutatakse välja eraldi käiguna. Aktiveerimise korral on olemas kolm režiimi (*Add*, *Subtract*, *Both*) ja energia parameeter *Energy* (mis muudab varjutamise intensiivsust). Kolmas muutuja näitab, kui palju kaudvarju materjal saab. Kui ei saa üldse, siis kaudvari seda ei mõjuta. Nende muutujate põhjal arvutab Blender välja AO käigu. Kui on soov seda eraldi renderduskäiguna kasutada ja siis see pilt komposiitorisse suunata, tuleb võimaldada ka Color- ja Diffuse-renderduskäik.

Nuudelduse paikapaneuks võid kasutada ülalolevat näidispiilt.

1. . Sõltuvalt AO režiimist tuleb teha järgnevat: Kui AO režiimiks on lisamine, siis asuta AO käiku otse. Kui AO režiim on *Sub*, siis arvuta $AO - 1$, või kui AO mode on *Both*, siis arvuta $2 * AO - 1$
2. . Esimese sammu väljund korruta (*Mix*; *Multiply* sõlm) AO energiatasemega
3. . Teise sammu väljund korruta materjali *Ambience*-väärtusega. Kui on kasutusel erinevad üldvalguse väärtust vastu võtavad materjalid, tuleks teha Object ID-l põhinev *Ambience*-kaart.
4. . Kolmanda sammu väljund korruta *Color*-käiguga
5. . Neljanda sammu väljund lisa käigule *Diffuse*

Kui varjud, värviline üldvalgus, läige, peegeldused ja/või murdumised on kaasatud, tuleb need lisada diffuse-käigule enne teisendatud AO käiku.

Vektorhäguga varjud



Liikumishägu (*Motion Blur*) asemel vektorhägu sõlme (Vector Blur) kasutamisel on nii, et liikuvad objektid on hägusad, aga kaamera suhtes mitteliikuvad objektid ei ole. Probleemiks on siin liikuva objekti vari. Ülal on liikuv kuubik tasandi kohal. Kui stseen lihtsalt läbi vektorhägu sõlme lasta, on tulemuseks see, et kuubik on liikumise suunal hägune, kuubiku vari aga selgepiiriline. Pildil on see näha paremal all servas.

See tähendab, et tuleb teha üks täiendav "põranda" nimeline renderduskiht, milles on eraldatud renderdamise käikudest hajumine ja vari (*Diffuse, Shadow*). Selle kihi puhul on võimaldatud *Diffuse* ja *Shadow* käigud ning see renderdab ainult põranda (kiht 2). Teine renderduskiht ("Kuup") renderdab Z ja *Vector* käigud ja ainult kuubi (kiht 1). Hägustamissõlmega (*Blur*) tuleb hägustada varjude käik ja siis ühendada see hajunud ja hägustatud varjuga ja siis *Multiply* sõlmega kokku miksida; tulemuseks on hägune vari selge tasandi peal. Lõpuks võib realistliku pildi saamiseks selle vektorhägustatud pildiga kokku miksida.

Kokkuvõte

Renderduskäigud, mida Blenderist 2.43 alates otse torkida saab, annavad pea täieliku kontrolli lõpptulemuse üle. Objektide poolt tekitatud vari, mis pole nende tõeline vari, objektide fookusest välja ajamine või fookusesse sättimine päris kaamera mulje jätmiseks või puhtalt aja kokkuhoiu eesmärgil renderduskäikude muutmine - kõiki neid võib olla vaja renderdusmootoris seadistada.

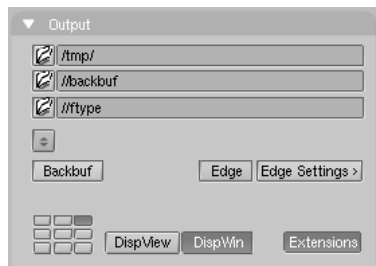
Serva renderdamine (*Toon Edge*)



Stseen multifilmistiilis.

Blenderi varjutaja *toon* abil saab koomiksilaadse või manga-stiilis väljanägemise, pakkudes kahetoonilist varjutamist. Päris perfektne see efekt siiski pole, sest päris koomiksile ja mangade puhul kasutatakse käsitsi joonitud servasid. Blenderis on see võimalus järeltöötlusena olemas.

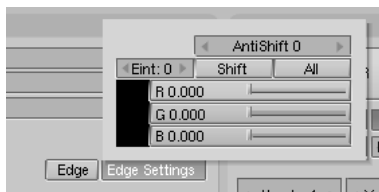
Valikud



Multifilmiliku serva tekitamise nupud.

Edge (serv)

Selle sisselülitamisel otsib Blender üles objekti servad ja "joonistab" need üle.



Multifilmiliku serva tekitamise nupud.

Enne kordusrenderdust on vaja mõned parameetrid paika panna:

Threshold (lävi)

Kui suurte külgede kaldenurga korral jooni tegema hakata, väärtusskaalal 0 kuni 255. Number 10 annab serva ainult sinna, kus on taust. Suurem väärtus hakkab serva joonistama juba vähem terava nurga all olevatele servadele. Kõige suurema väärtuse puhul tekivad servad isegi väga sujuvate geomeetriajoonte juurde.

Kui kõige suurema numbri sisestamisel tekkinud tulemusest jääb väheks või kui kõige väiksema numbri sisestamisel on joon ikka liiga märgatav, saab tulemust muuta asjade mõõtkava muutmisega. Näiteks võib vähemmärgatava joone saamiseks stseenil olevaid objekte (ja kaamerat) proportsionaalselt suurendada.

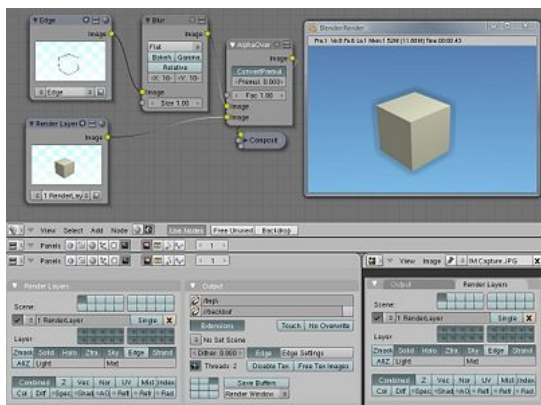
Colour / R/G/B (värv)

Renderdatud serva värvitoon (vaikimisi must). Värv valimiseks klõpsa värvi valimise tööriistal.

Examples



Multifilmiliku servaga uuesti renderdatud pilt



Serva saamiseks saab teha eraldi, just selle jaoks mõeldud kihi. Alfakanalis pole serva juhul, kui läbipaistvuskanali väärtus on null, ja serv on olemas siis, kui läbipaistvuskanali väärtus on 1. Servakihi eraldi tegemise korral on võimalik seda hägustada, muuta selle värvi, seda maskida jne. Parempoolset pildilt näed, kuidas seda teha. Tekitasin serva jaoks renderduskihi, milles on ainult taeva kiht ja serva kihid (ma jätsin taeva sisse, nii et me saame maailma värvid hiljem komposiitmisel). Teine renderduskiht serva kihti ei kasuta ja nii tuleb välja ainult tavaline pilt. Output-paneelil lülitan sisse 10 ühiku paksuse musta serva. Saadan selle kihi läbi hägustamise sõlme. Pärast seda kasutan *AlphaOver*-sõlme kuubiku hägustatud serva peale panekuks. Tulemuseks on pehme servaga varju efekt. Pane tähele, et Premultiply on valitud selle pärast, servaga kujutaval pildil on juba määratud läbipaistvuseks 1.

Virvtoonimine (*Dithering*)

Virvtoonimine on pikslite hägustamine nii, et värviüleminekualades ei tekiks toonide vahel "astmelist" üleminekut. "Astmeliseks minek" paistab silma pikemate värvisulatuste puhul või siis, kui värvi üleminek on väga õrn. Virvtoonimine leiutati seleks, et saaks vähesel bitisügavusega graafika puhul "rohkem toone" kasutada.

Virvtoonimine toimub nii, et mõõdetakse piksli värv, võrreldakse seda naaberpikslite ja piirväärtusega ja siis määratakse sobiv värv. Virvtoonimise abil jäetakse mulje suuremast saadaolevast värvide arvust. Näiteks punaste ja kollaste pikslite ühtlase jaotamisel tundub pind kaugemalt vaadates oranžina.

Dither ehk virvtoonimise väärtust saab määrata vahemikus 0 kuni 2.

Tempel (*Stamp*)

Stamp-paneelilt saab valida "templi" lisamise seadeid. Selle abil saab renderdatavale pildile peale panna teksti, mis võib sisaldada järgmist informatsiooni:

Time (aeg)

Kuva sõna *Time* ja selle järel eesoleva stseeni aeg kujul HH:MM:SS.FF.

Date (kuupäev)

Kuva renderdamise hetkel olnud aeg ja kuupäev.

RenderTime (renderdusaeg)

Kuva eesoleva kaadri renderdamisele kulunud aeg.

Frame (kaader)

Kuva kaadri järjekorranumber.

Scene (stseen)

Kuva aktiivse stseeni nimi.

Camera (kaamera)

Kuva filmiva kaamera nimi.

Lens (läätis)

Kuva filmiva kaamera läätse fookuskaugus.

Filename (failinimi)

Kuva .blend-faili nimi.

Marker

Kuva viimase markeri nimi.

Seq. Strip (videoriba)

Lisa esiplaanil oleva videoriba nimi.

Note (märkus)

Lisa enda poolt kirjutatud tekst.

Stamp Text Color (templi teksti värvus)

Teksti värvi ja läbipaistvuse määramine.

Stamp Background (templi taust)

Teksti taustavärvi ja läbipaistvuse määramine.

Font (kirjatüüp)

Teksti suuruse määramine.

Liikumishägu (*Motion Blur*)

Vaikimisi renderdab Blender animatsioone perfektelt selgete staatiliste piltidena. Täiesti selged pildid tunduvad aga ebarealistlikud, sest objekti enda liikumine muudab selle nii vaataja kui ka kaamera jaoks liikumise suunal häguseks. Liikumishägu efekti saavutamiseks renderdab Blender hetkekaadri ette ja taha vahekaadrid ja siis sulatab need üheks pildiks kokku. Nii paistavad liikuvad objektid liikumise suunal udused ja jääb mulje liikumisest.

Inimsilm

Inimese aju suudab töödelda umbes 15 pilti sekundis, mõlemas silmas paralleelselt. Ajus sulanduvad järjest kuvatavad pildid ja nende erinevust tajume liikumisena. Kui miski liigub piisavalt kiiresti, eeldan seda nägevat häguna (kas siis selle pärast, et silmas olevad "kepikesed" reageerivad valgusele viivitusega või siis koostab aju liikuvatest piltidest häguse liikuva). Mõte on selles, et ma "tajun" liikumishägu olemasolu.

Filmis

Katkendliku liikuva pildi vältimiseks kahekordistati kaadrisagedust 30 kaadri sekundis (24 kaadrit Euroopas). Põhimõtteliselt on katik avatud 1/30 sekundit ja sama kaua seda kaadrit ka säritatakse. Kui säritamise ajal miski liikus, jääb see kaadrisse hägusena või laialimäärituna. Pärast ilmutamist on näha hägust pilti. Mõte on selles et ma "näen" liikumishägu olemasolu.

Arvutigraafikas

Arvutigraafikas teab arvuti täpselt, kus miski peaks paiknema, ja renderdab selle vastavalt. Kaadrist kaadrisse on objekt kaadris 1 kohas A ja kaadris 2 kohas B. Kaadrisagedusel 30 kaadrit sekundis (*fps*) tundub pilt meile hüplik, sest puudub seda ühtlasemaks muutev hägu, nagu see on päris filmi puhul.

Liikumishägu Blenderis

Kas ka arvutigraafika abil saab liikumishäguga pilte teha? Blenderis on liikumishägu saamiseks kaks moodust:

Sämplitud

See on aeglane, aga samas parema tulemusega. Selle saab sisse lülitada liikumishägu (Motion blur) sektionist renderdamise seadistuste paneelil.

Motion Samples (liikumissämplid)

Määrab, mitmest kaadrist aktiivne kaader kokku sulatu.b

Shutter (katik)

Aeg kaadrites katiku avanemise ja sulgumise vahel.

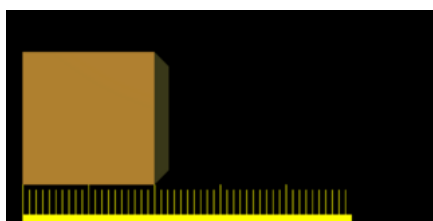
Vektorhägu

[Vektorhägu \(Vector Blur\)](#) on kiirem, aga mõningate kõrvalmõjudega, mida saab siiski vältida. Vektorhägu arvutatakse komposiitmisel. Renderdamisvalikute all tuleb Vector Pass sisse lülitada. Nii talletatakse iga piksli liikumisinformatsioon. See info on vektorsuurus, mis kirjeldab liikumise 2D- või 3D-suunda ja pikkust. Komposiitor kasutab seda infot piksli soovitud suunal hägustamiseks.

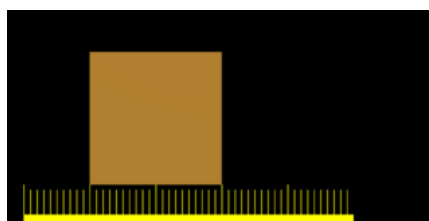
Näited

Asja olemusest arusaamiseks oletame, et meil on vaikekuubik, mis liigub ühtlaselt 1 Blenderi ühiku võrra iga kaadri kohta paremale. See on üsnagi kiire, sest kuubiku enda mõõtmed on 2 Blenderi ühikut.

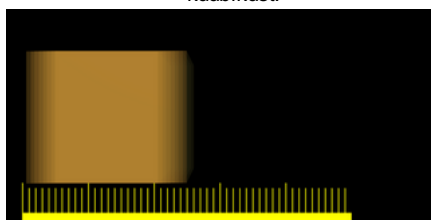
Pilt 1 näitab esimese kaadri renderdust ilma liikumishäguta, *Pilt 2* näitab kaader 2 renderdust. Kuubiku all olev skaala aitab hinnata kuubiku liikumist 1 Blenderi ühiku võrra.



Pilt 1. Kaader 1 liikumishäguta liikuvast kuubikust.



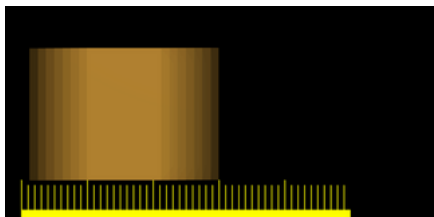
Pilt 2. Kaader 2 liikumishäguta liikuvast kuubikust.



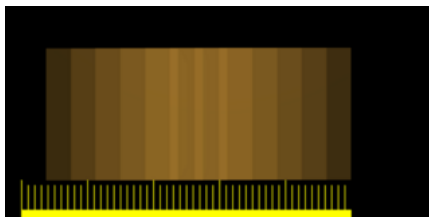
Pilt 3. Kaader 1 liikuvast kuubikust liikumishäguga, 8 sämplit, säriaeg=0,5

Pilt 3 näitab kaadri 1 renderdamist nii, et liikumishägu sämpleid on 8. Katik (Shutter) on 0.5; see tähendab, et on arvutatud 8 kaadrit perioodiga 0,5 kaadrist 1. Hägusus on väga ilmne, sest pool hägu on ühel ja teine pool teisel pool põhikuubikut.

Pildil 4 ja Pildil 5 on näidatud Bf-väärtuse suurendamist. Suurus üle ühe tähendab väga "aeglast" katikut.



Pilt 4. Kaader 1 liikumishäguga liikuvast kuubikust, 8 sãmplit, katik = 1,0.



Pilt 5. Kaader 1 liikumishäguga liikuvast kuubikust, 8 sãmplit, katik=3,0.

Näidisest paremad tulemused saab siis, kui panna näidises olnud 8 asemele 11 või 16 sãmplit. Iga täiendav eraldi renderdus võtab aega, kuid parandab liikumishägu kvaliteeti.

Vihjeid

Liikumishägu sisselülitamisel "võdistab" Blender kaamerat eelmise ja järgmise kaadri vahel isegi siis, kui midagi ei liigu. See tähendab, et isegi sakisilumise väljalülitamisel on saadud pildid ilusasti silutud. Liikumishäguga saadud sakisilumine on "päris" sakisilumisega võrreldav, kuid see on üldiselt aeglasem.

Seda on huvitav teada, sest väga keeruliste stseenide korral ei pruugi ka 16-segmendiline sakisilumine rahuldavat tulemust anda. Kuid korraga nii sakisilumise kui ka liikumishägu sisselülitamine annab parema tulemuse. Sel viisil saab olla sãmpleid niipalju, kui on stseenis üksteisele järgnevaid kaadreid. Nii saab sakisilumise sãmplite arv olla 25, 64, 121, 256, kui väärtus on vastavalt 5, 8, 11, 16.

Kirjeldus

Blenderisse sisse ehitatud Võrgurenderdaja

Eesmärgid

- Läbipaistvus
- Kohaldatavus

Allalaadimine

Versioonis v2.5 (svn tüvest)

Uusimate võimaluste ja värskeimate veaparanduste saamiseks on tungivalt soovitatav kasutada uusimat SVN-versiooni. Muul juhul on soovitatav kasutada ametlikku väljalaset (hetkeseisuga v2.56).

Juhised

Kasutajaliides

Isand (*Master*)

Ühe masina peal käivita peaserver.

- Käivita Blender ja vali ülemiselt ribalt Blender Render asemel renderdusmootoriks Network Render. (Selle valiku nägemiseks on vaja see kõigepealt järgmises kohas sisse lülitada: File » User » Preferences » Add-Ons » Render)
- Vali töörežiimiks *Master*
- Soovitatav on kirja panna nii võrguliidese IP-aadress kui ka port. Jäta see vaikimisi *[default]* seadistuseks juhul, kui tahad, et arvuti kuulaks kõiki määratud pordi võrguliideseid.
- Vajuta *Start* nuppu (selle tulemusel avaneb tühi renderdusaken). Nüüd näitab renderduse indikaatorjoon serveri tegevusi
- Serveri töö saab lõpetada Esc-klahvile vajutades. Täpselt nii, nagu saab katkestada tavalise renderdamise.

Orjad (*Slaves*)

Teistes arvutites käivita NetRender orjarežiimis (*Slave*)

- Käivita Blender, lülita renderdusmootor *Network Render* peale
- Vali toimetusréžiimiks *Slave*.
- Veel on soovitatav ära märkida peaserveri IP aadress ja port. Kasuta vaikimisi *[default]* valikut sel juhul, kui sa tahad et *Master* arvuti tuvastamine toimuks automaatselt.
- Vajuta *Start* nuppu (selle tulemusel avaneb tühi renderdusaken). Renderdamise indikaatorriba näitab orja aktiivsust
- Ori töötab seni, kuni peatamiseks Esc-klahvi vajutatakse. Täpselt nii, nagu tavalisegi renderdamise korral.

Klient

Renderdustöö saatmiseks oma tööarvutist klastrisse:

- Ava renderdatav .blend-fail. Kontrolli üle renderdusseadistused (suurus ja muu)
- Salvesta fail (selle tulemusel saadetakse teele viimane versioon)
- Vali renderdusmootoriks *Network Render*
- Vali töörežiimiks *Client*
- Tee järgnevatest üht:
 - Määra ära põhiserveri IP-aadress ja port.
 - Peaserveri automaatseks avastamiseks vajuta *Refresh* nuppu aadressiriba all.
- Töö lähetamiseks vajuta nuppu *Send Job*.
- Animatsiooni renderdamiseks vajuta nuppu *Animation* (Ctrl-F12). Lõpuni renderdatud kaadrid jõuavad automaatselt tagasi ja saatmise ajal toimub väike paus.
- Eesoleva üksikkaadri renderdamiseks võib vajutada nuppu *Render* ja see saadetakse põhjarvutisse renderdamiseks.
- Lihtsaima näitena võid lihtsalt vajutada nuppu *Animation on network* ja siis kaadrite saabumist oodata. Renderduse koguaeg peaks olema proportsioonis ori-arvutite hulga (miinus failide edastamisele kulunud aeg).

Käsurida

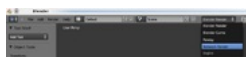
- Seadista peaserver ja orjad nii, nagu eespool kirjeldatud. Salvesta konfiguratsioon eraldi .blend-failidesse (näiteks: master.blend ja slave.blend).
- Taustarenderduse jaoks käivita peaserver ja orjad järgmiselt:
 - *blender -b master.blend -a*
 - *blender -b slave.blend -a*
- Isandat ja orja ei saa peatada klahvikombinatsiooniga Ctrl-C (enne peaserveri peatamist on soovitatav peatada orjad).

Lisainfo

Renderdamisel kasutatakse täielikku mitmekihelist renderdust, nii et lõpptulemus on sama nagu kohalikus arvutis renderdamisel. Selle saamiseks ei pea sa originaalfailis seda eraldi ära määrama. Tööarvutites tehakse seda automaatselt.

Testijad on oodatud minuga ühendust võtma kasutajaga **theeth** [IRC \(#blendercoders\)](http://irc.blendercoders.com). Ühendust võib võtta ka e-kirjaga.

Seadistused



NetRender

renderdusmootorina

Renderdusmootori valiku menüü on Blenderi ülaseri keskosas. Sellest kohast saab valida renderdusmootoriks *Network Render*.

Peaserver



Peaserveri seaded

- **Network Settings**
 - **Start Service:** käivitab peaserveri
 - **Path:** näidatakse ära, kuhu peaserver salvestab töös olevad failid, lõplikud failid, logid jm failid. See tekitab uue kausta kujul *master_<pid>*, kus *<pid>* on peaserveri protsessi ID.
 - **Server Address:** peaserveri poolt kuulatava võrguliidese aadress. *[default]* ehk vaikimisi tähendab, et kuulatakse kõik võrguliideseid.
 - **Port:** millist porti peaserver kuulab.
 - **Open Master Monitor:** avab brauseris peaserveri monitoorimise lehe. Peaserveri töötamise ajal on see sisse lülitatud.
- Peaserveri seaded
 - **Broadcast:** edastab kohtvõrku info peaserveri aadressi ja pordi kohta (iga 10 sekundi tagant).
 - **Clear on exit:** kustutab peaserveri peatamisel väljaga *Path* määratud kataloogi.

Ori



Orja seadistused

- Võrguseaded
 - **Start Service:** käivitab orja sõlme.
 - **Path:** määrab orja poolt kasutatav kaust tööfailide, lõppfailide ja logide salvestamiseks. See tekitab uue kausta kujul *slave_<id>*, kus *<id>* on peaserveri poolt antud orja ID-number.
 - **Server Address:** millist aadressi peaserver kuulab.
 - **Port:** millist porti peaserver kuulab.
 - **Refresh:** kuula peaserveri poolt saadetud signaali, et saaks aadressi ja pordi kindlaks teha (selleks võib kuluda kuni 20 sekundit).
 - **Open Master Monitor:** avab brauseris peaserveri monitoorimise lehe. Seda saab kasutada juhul, kui peaserveri aadress on õige.
- Oriarvuti seaded
 - **Clear on exit:** kustutab pärast orja peatamist määratud kataloogi.
 - **Generate thumbnails:** tekitab renderduste pispildid oriarvutis (vastasel korral teeb seda vastavalt nõudlusele peaserver).
 - **Threads:** mitut protsessori tuuma peab oriarvuti renderdamiseks kasutama.

Klient



Kliendi seadistused

- Võrguseaded
 - **Path**: määratakse kliendi poolt ajutiste renderduste jaoks kasutatav fail.
 - **Server Address**: millist aadressi peaserver kuulab.
 - **Port**: millist porti peaserver kuulab.
 - **Refresh**: kuula peaserveri poolt saadetud signaali, et saaks aadressi ja porti kindlaks teha (selleks võib kuluda kuni 20 sekundit).
 - **Open Master Monitor**: avab brauseris peaserveri monitoorimise lehe. Seda saab kasutada juhul, kui peaserveri aadress on õige.
- Töö seadistused
 - **Animation on network**: saadab praeguse faili peaserverisse uue ülesandena ja ootab tulemusi (kui välja arvata, et renderdamine toimub teises arvutis, on tegemist täiesti tavalise animatsiooni renderdamisega).
 - **Send Job**: saadab käsiloleva faili renderdustööna peaserverisse. Tagasi saadetud töö ID-st saab *käsiloleva töö ID*.
 - **Get Image**: nõuab peaserverist käsiloleva töö parajasti arvutatavat kaadrit (kui see pole veel valmis, ootab valmimiseni). Tulemuseks on tavaline renderdatud kaader.
 - **Get Animation**: küsib peaserverist kõiki käsiloleva töö kaadreid (kui mõni kaader pole veel valmis, siis ootab valmimiseni). Tulemus on sama mis tavalise renderdatud animatsiooni puhul.
 - **Name**: renderdustöö nimi. *[default]* kasutab nime .blend-faili nime.
 - **Category**: töö kategooria, *soovi korral* lisatav. Peaserver tasakaalustab tööde teostamist vastavalt kategooriale.
 - **Priority**: töö ettevõtmise tähtsuse järjekord. Prioriteeditaseme kordaja paneb peaserveri ülesannet arvestama nii mitme tööülesandena kui kordaja määrab (näiteks priority 1 saab 33% ja priority 2 66% saadaolevast tööajast).
 - **Chunks**: kui mitu kaadrit orjale korraga saadetakse.
- Orjade olek
 - **List**: näitab ära kõik peaserveriga ühendatud orjad.
 - **Refresh**: värskendab peaserverist saadavat infot orjade kohta
 - **Remove**: paneb vastava orja musta nimekirja.
- Orjade must nimekiri
 - **List**: kõikide musta nimekirja pandud orjade nimistu.
 - **Remove**: eemalda vastav ori mustast nimekirjast.
- Tööd
 - **List**: kõikide peaserveris olevate tööde nimekiri.
 - **Refresh**: värskendab infot peaserveris olevate tööde kohta
 - **Remove**: eemaldab peaserverist vastava töö.
 - **Remove All**: eemaldab peaserverist kõik tööd.
 - **Get Results**: nõuab endale kõik valitud töö saadaolevad kaadrid. Failid laetakse mitmekihilises EXR-formaadis käsilolevasse väljundkataloogi.

Märkused ja teadaolevad vead

- Sõlmedele pole jagatud võrguketast tarvis
- Korraga võib töösse anda palju faile, tulemused saab kätte sõltumatult (kui tahad faili hiljem tagasi saada, salvesta see enne sulgemist).
- Võrguvigade veahaldus peaaegu puudub. Kui paned peaserveri esimesena kinni, lähevad asjad katki. Sama juhtub vigase aadressi korral.

Praegune teostus on teadagi ideaalsest kaugel. Eriti kui vaadata seda professionaalse renderdusfarmi seisukohast. Loodan, et Matt sunnib mind tagant ja soovitab paremaid lahendusi. Hea oleks variant, kus kasutaja peaks kõigest "Anim on network"-nuppu vajutama ja kõik tehakse ära nii, nagu see oleks tavaline kohalik renderdus. Kõigi professionaalsete valikute tegemine peaks olema vabatahtlik.

Koormuse tasakaalustamine

Esimene tasakaalustamine tehakse nii, et keskmistatakse iga klatri kasutus konkreetse töö kohta iga 10 sekundi tagant. Järgmisena antakse töösse kõige vähem töös olev töö (mis kasutab kõige vähem orje). Järgmisena toimub töö jaotamine prioriteedinumbri alusel. Prioriteediga 2 olev töö saab enda käsutusse kaks korda suurema arvu orje kui prioriteediga 1 olev töö. Peale selle on prioriteedi jaoks veel mõningaid erandeid ja reegleid:

Erandid

- Üksik töö ei saa kasutada rohkem kui N% kõikidest orjadest, v.a siis, kui see on ainus töö. Selle abil saab vältida seda, et kaua aega võttev töö takistab lühikesi töid. Praegu on see pandud 75% peale, aga seda saab muuta.

Esmased prioriteedid (kriteeriumid)

- Kui on saadetud alla N kaadreid (uute tööde prioritseerimine). Selle mõte on selles, et nii saab vigadele varem jälile.
- Rohkem kui N minutit viimasest lähetamisest. Et kõrge prioriteediga tööd teisi ära ei näljutaks.

Mis on veel plaanis teha

- Töö saatmine mälust
- Et nähtavus ei sõltuks valitud renderdusmootori valikust
- Pädevam renderduse haldaja ekspertidele
- Paremini määratletud sideprotokoll
- Valik simulatsioonide arvutamiseks (riided, toss jne), nii et vastavat asja tegevad sõlmed saaksid oma vahemälu renderdavas serverisse saata.
- Üleslaadimisel automaatne tekstuuride kaasapakkimine
- Ühe kaadri osadeks jagamine ja iga osa eraldi tööks saatmise võimalus

Tehnilised täpsustused

See on aegunud. Loe koodi ja pane info sinna.

Võimaluste loetelu

- failide asemel failiradade toetus
- client-server-slave: piira tehtava töö jaotamine vaid kindlatele sõlmedele
- client-server-slave: vaata sõlmede seisukorda
- client-server-slave: vealogid lähevad haldurisse tagasi (sõlmedest `stdout` ja `stderr`)
- Töö katkestamine
- Vigase kaadri kohalt jätkamine
- Vearaporteerimise väljalülitamine Windowsis
- Rohkem kui 1 kaadri saatmine (kaadrite ulatus)
- Orja musta nimekirja saatmine, kui see annab veateadet ka pärast taaskäivitamist
- Ühes töös mitme raja kasutamine
- Tööga viivitamine seni, kuni info kõigi failide kohta on käes
- Kaadri vahemikele piirangute seadmine (näiteks punktide vahemälu saatmine ainult vajalike kaadrite korral)
- Osaliste logide peaserverisse saatmine
- TEGEMISEL: Orjade sundimine tulemuste kopeerimiseks võrgurajale
- TEGEMISEL: client-master: tööde archiveerimine (lähtefailide ja saadud failide kopeerimine)
- TEGEMISEL: master-slave: tööde piiramine vastavalt orjade tehnilistele andmetele.

API funktsioonide soovinimekiri

Netrenderi paremaks muutmiseks oleks vaja selliseid koodijuppe. Mõned neist on vead, mõned neist aga funktsioonid, mis kunagi siia peaksid jõudma.

- API ligipääs töödele, et oleks võimalik peaserveri ja orjade taustal käitamine ning klientidele teavituste saatmine.
- Mitmekihilises formaadis tulemuse mälu talletamine
- Renderdamiseks renderdussegmentide laadimine

Subpages

- [Yafaray](#)

Kirjeldus

Yafaray, nagu selle pikem nimi ütleb, (*Yet Another Free RAYtracer* Veel Üks Vabavaraline Kiirtejälitaja), on tasuta, XML-keelt kasutav, platvormiülene kiirtejälitaja, mida arendab [Yafaray meeskond](#). See töötab paljude 3D-modelleerimisprogrammidega (nagu näiteks Wings ja Aztec), aga käesoleva kirjutise eesmärk on tutvustada ainult kasutamist Blenderiga.

Yafaray on saadaval ([LGPL litsentsi alusel](#)) Windowsi, Linuxi (lähtekoodina, .deb või .rpm installatsioonipaketina) ja Mac OSX-i Intel versiooni jaoks. Yafaray lähtekoodi saab alla laadida [siit](#).

Valikud

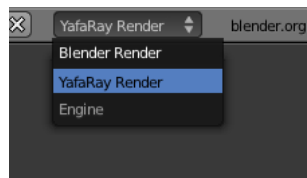
Alates Blenderi versioonist 2.34 kuni 2.49b saab seda kasutada sisese renderdusmootori asemel (eeldame, et see on installeeritud). Blender 2.5x versiooni jaoks veel eraldi installeeritavat versiooni ei pakuta. Graphical.org lehelt on võimalik alla tömmata sisseehitatud Yafarayga Blenderi versioone. Blenderi ülaserava keskosas on rippmenüüs vaikselt näha Blender Render. Yafaray kasutamiseks vali seal Yafaray. Kui kasutajaeelistuste (*User Preferences*) all on sisse lülitatud vaid üks renderdaja, siis seda rippmenüüd näha ei olegi. Mine *File/User Preferences/Add-Ons/Render* ja lülita sisse vähemalt kaks renderdajat, et menüü nähtavale ilmuks.

Render Pipeline

Yafaray kasutamise korral muutub see üheks osaks töövoost (*workflow*) enne komposiitorit ja järjestajat. Seda seetõttu, et komposiitor ja järjestaja kasutavad renderdatud pilte. Yafaray renderdajale antakse info stseenil olevate objektide, materjalide, valgustite ja muu kohta. Yafaray ei arvesta renderduskihtide olemasoluga. Selle väljundit ei saa suunata komposiitorisse või järjestajasse, sest see toimib Blenderi sisemisest renderdusmootorist erinevalt. Yafaray renderdab kaadrid Blenderi stseeni järgi.

Yafaray poolt renderdatud pilte saab komposiitoris kasutada nii, et tuleb kasutada pildi sisendi sõlme (image input node). Seejärel saab selle ribana järjestajasse tuua ja komposiitori sisse lülitada. Yafaray väljundi võib pärast renderdamist otse järjestajasse sõota pildiriba (*Image strip*) abil.

Pärast renderdusmootori rippmenüüst Yafaray valimist tekib renderdamise sakile veel 2 lisapaneeli. Nendelt paneelidelt saab muuta mitmeid Yafaray seadeid. Paneelil mitteolevaid parameetreid saab muuta läbi XML koodi.



Yafaray lubamine

XML

Sellele nupule vajutamise järel eksporditakse stseen enne Yafaray poolt renderdamist XML-faili "tmp" kataloogis. See on kasulik siis, kui on tarvis teha muudatusi XML-failis või renderdada käsurea abil. Ära seda vali, kui sa tahad visuaalselt renderdamisprotsessi edenemist Blenderi aknas vaadata.

AutoAA

See võimaldab lülitada automaatse ja käsitsi sakisilumise vahel. Yafaray sakisilumine toimib sarnaselt Blenderi enda omaga.

Mõningatel juhtudel on vaja käsitsi muuta sakisilumise seadeid (näiteks kui on plaanis Yafaray sügavusteravust kasutada). Kasulik on meelde jätta, et segmentide arvu suurendamine käigu (*pass*) kohta suurendab lõpliku renderdatud pildi peal servade täpsust; vähendamine jällegi vähendab täpsust ja servad näevad välja karvased ja treppis.

Proc.

Siit saad valida, mitut protsessorit Yafaray kasutada saab. Ühe protsessoriga arvutite puhul pole vaja seda muuta.

Gam.

Selle abil saab käsitsi stseeni gamma väärtust muuta. Vaikeväärtus 1 lülitab selle välja.

Exp.

Selle abil saab ise stseeni säritust määrata. Põhjalikum seletus on hilisemas osas.

Üldvalgustus



Globaalse valgustuse seadistused

<vananenud info>Järgmine sakk "Yafaray GI" sisaldab Yafaray valgustamise meetodite valikuid.</vananenud info> Saadaval on sellised variandid:

Full (täielik)

See sobib enamiku stseenide jaoks, kuid rohkem siiski siseruumi stseenide jaoks, kus footonite kasutamine on asjakohasem.

Skydome (taevakuppel)

See variant on sobilik "õues" olevate stseenide jaoks.

Cache (puhver)

Vahemälu kasutamine aitab Yafarayl renderdamist kiirendada, sest siis saab renderdaja segmentide jaotamisel valivam olla.

Kui see nupp on väljalülitatud olekus, arvestab Yafaray eraldi käiguna välja kõige sobivamad sampilte asukohad ja alles siis asub pilti ennast renderdama. Nii tõuseb efektiivsus.

Cache nupu vajutamine toob nähtavale veel kolm nuppu.

ShadQu (varju kvaliteet)

Sellega saab muuta varju kvaliteeti. Suurendades seda sätet vaikeväärtusest (0.900) kõrgemaks, tõstad sa varjudega alades võetavate samplite ehk proovide arvu, mis omakorda suurendab stseenis olevate varjude kvaliteeti, kuid pikendab ka renderdusaega.

Prec (täpsus)

See säte määrab ilma sampliteta ruudu maksimaalse pindala pikslites. Vähendades seda sätet vaikeväärtusest (10) madalamaks, tõstad sa stseenis võetavate samplite ehk proovide arvu. Selle sätte vähendamine pikendab renderdusaega.

Ref (täpsustus)

Selle sätte abil saad määrata varjude täpsemaks muutmise läve. Vähendades seda sätet vaikeväärtusest (1.000) madalamaks, palud sa Yafarayl suurendada käikude arvu, mida varjudega alades samplite jagamiseks kasutatakse. See omakorda suurendab stseenis olevate varjude kvaliteeti, kuid pikendab ka renderdusaega.

Näited

Alusta Blenderi vaikeseadistustega, vali renderdusmootoriks Yafaray, muuda XML mittemärgituks. Vajuta nuppu "Render" (F12).

Konsooli väljund

Kui keskkond seda lubab, peaks Yafaray väljastama informatsiooni konsooliaknasse (Windowsis avab Blender endaga koos ka konsooliakna. GNU/Linuxis aga tuleb konsooliväljundi nägemiseks avada Blender konsooli kaudu – tavaliselt piisab terminali emulaatori aknasse "blender" trükkimisest).

Kui sa vaatad konsooli pärast renderduse lõppu, peaksid sa (kui puhverdamise säte "cache" on sisse lülitatud) nägema midagi järgnevat:

Konsooli väljund

Launching 1 threads

Fake pass: [#####]
534 samples taken

Render pass: [#####]

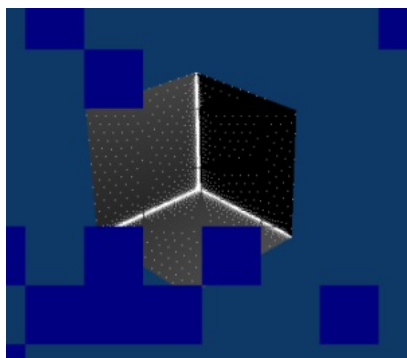
render finished

Väljundi kirjeldus

Renderdus on jagatud kahte eraldi käiku (*passes*). Esimene, "fake" ehk võltskäik tehakse puhvisätte "cache" sisselülitamise tulemusena ning selle eesmärgiks on välja selgitada parim samplite jagunemine stseenis (ilma puhverdamissätteta jagatakse samplid stseenis ühtlaselt). Seejärel näidatakse järgmisel real samplite arvu.

Järgmine käik on "real" ehk reaalne renderduskäik, mille jooksul Yafaray arvutab eelmises sammus loodud samplite laotuse tulemusena pildi.

Renderdamine ilma väljundita



Suurem samplite arv varjudega piirkondades

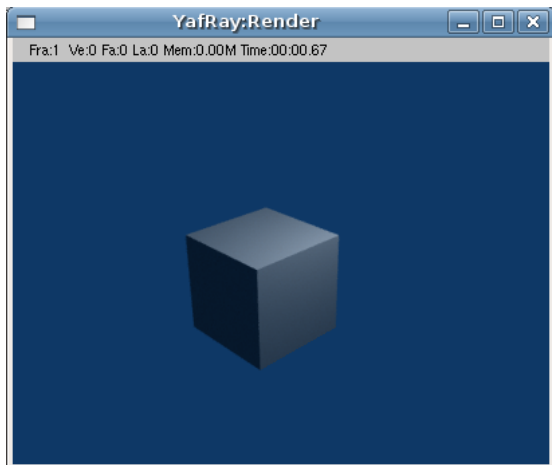
Nüüd vaatleme Yafaray väljundit renderdusaknasse renderdamise ajal.

Kui XMLi väljund on väljalülitatud, värskendab Yafaray pidevalt renderdusakna visuaalset väljundit – nagu Blender isegi seda teeb. Paremal olev pilt võeti renderdamise võltskäigu ajal ning valged punktid kujutavad samplite jagunemist stseenis. Pane tähele, et samplid on asetatud ainult nendesse stseeni piirkondadesse, mida valgus otseselt mõjutab. See tähendab, et näidispildil arvestatakse ainult nende stseeni osadega, millel on pinnad.

Samuti tähendab see seda, et varjulistes stseeni piirkondades on samplite arv suurem.

Pildilt näed, kuidas valgete punktide tihedus, mis, nagu ma viitasin, tähistab selles pildi alas ühe piksli kohta võetavate samplite arvu, on suurem piirkondades, millel on suurem tõenäosus olla varjus (antud juhul kustutasin ma kaamerale kõige lähemal oleva kuubi tipu

ning tõin sellega nähtavale sisemised servad, mis on vähem valguse käes).



Lihtne YafRay renderdus

Renderdatud pilt

On näha, et Blenderi vaikekuubiku vaikimisi hall materjal on sinine. See on nii selle pärast, et täielik (*Full*) variant mõjutab stseeni värve maailma värvidega, mis on Blenderis vaikimisi sinine. Selle muutmiseks mine varjutuse paneelile *Shading* (F5) ja vali sealt maailma ikoon. Et materjalide välimus oleks õige, määra maailmavarjutaja värviks valge.



Maailmavarjutaja valimine

Märkused

Valguse hulk



YafRay käitub valgusega Blenderi sisemisest renderdajast täiesti erinevalt ning seetõttu näib, et YafRay jaoks tuleb valguste intensiivsust kõvasti kõrgemaks tõsta. Pildid näitavad Blenderi sisese renderdaja, ilma globaalse valgustusega (*Global Illumination*, *GI*) YafRay renderduse ja täieliku globaalse valgustusega YafRay renderduse erinevusi. Nagu sa näed, on tulemused sõltuvalt valitud valgustamise meetodist väga erinevad.

Lahendusena võib kasutada väga suuri pindvalgusi (*Area lamps*, mis on ruudukujulised (*Square*), mille suurus (*Size*) on 100, aga sämplite arv (*Samples*) ainult 4 ja energia (*Energy*) 10), et saada pehmemaid varje. Selgete varjupiiride saamiseks aga koos

päikesevalgustiga (*Sun*), mille energia (*Energy*) väärtus on kõvasti väiksem (vähem kui 1.0). Päikesevalgustite intensiivsus näib YafRays olevat tükki maad suurem kui pindvalgustite oma, kuid varjude servad on suhteliselt teravad.

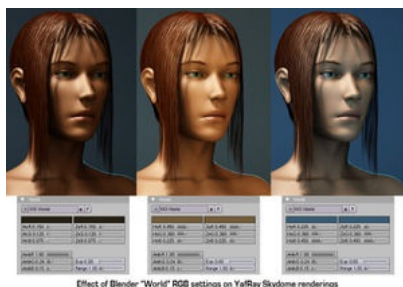
Proovi kasutada YafRay globaalse valgustuse jaoks taevakupli sätet *Skydome*, sest täieliku globaalse valgustusega võivad tekkida imelikud artefaktid, mida keegi ei näi oskavat parandada, kuid mis võivad olla seotud mu Blenderi stseeni mõõtkavaga (1 Blenderi ühik = 1 cm ning figuurid on elusuurused). Ka sina teed midagi sarnast, kui ehitad täpse mõõtkavaga mudeleid, mis vastavad kaamera perspektiivile.

Blenderi maailma parameetrid (*World*) võivad sisaldada väikest kaudvarju (AO) väärtust, millega YafRay näib arvestavat ning seetõttu proovi seda väheke ka oma stseenis kasutada. Samuti pane tähele, et maailma taeva (*World Sky*) värve (horisont ja seniit) käsitletakse poolkeravalgustina ning seetõttu toonivad nad su stseeni taevakuplit kasutades vastavalt – muuda natuke RGB-väärtuseid, et üldist intensiivsust globaalse valgustamise abil "täita". Allolevatel piltidel suurendati maailma valgustuse sätteid parempoolse renderduse jaoks kaks korda.

Kõik näib YafRay puhul vajavat suurendamist – mõned materjalid näevad väga tuhmid välja, kui sa mõningaid nende komponente ei "kahekordista" (näiteks kasutades piltekstuuri liitmisvalikuga *Add* kahel korral), ning *RGB* ja varjutaja *Shader* sakkide sätteid on väga erinevad sellest, mida sa sisemise renderdaja puhul kasutaksid.

Samuti võid muuta kiirgusvõimsuse (*EmitPwr*) ja säri (*Exp*) sätteid YafRay renderdamissakis, et valgustuse erinevusi kompenseerida. See võib olla suhteliselt keeruline. Boonusena saad sa aga stseeni tükki maad rikkalikuma iseloomuga valgustuse, mis võib nähtud vaeva väärt olla.

Taevakuppel (*SkyDome*)



Effect of Blender "World" RGB settings on YafRay Skydome renderings

Erinevad maailma (*World*) sätetel põhinevad toneerimiseffektid

Blenderi sisese renderdaja kasutamisel on ainus võimalus panna maailma horisondi (*Horizon*), seniidi (*Zenith*) või tekstuuri (*Textured*) mõjutama materjali värvi kaudvarju (*Ambient Occlusion*) kasutamine, mis on määratud taeva värvi (*Sky Color*) või taeva tekstuuri (*Sky Texture*) peale. Muidu (ilma kaudvarjuta) mõjutab see ainult tausta värvust. Ainus muutuja, mis otseselt Blenderi sisemises renderdajas mõjutab objekti värvust, on keskkonna valgus (*Ambient light*) ning sellisel juhul saab igale materjalile määrata selle keskkonna valguse hulga, mis teda mõjutab (vaikimisi 50%). Keskkonna valguse värv ei saa sisese renderdaja puhul pildi kõrguse ulatuses varieeruda ning see määratakse tegelasele ühtlaselt. Kaudvari (sõltuvalt sätetest) mõjutab mudeli värvi sõltuvalt selle geomeetriast.

YafRay puhul on peamiseks erinevuseks aga see, et kõigi nende sätete värvid on olulised (nagu näitepildilt on näha). Näitepildil on sama materjali (nahk ja juuksed) renderdatud erinevate **horisondi ja seniidi** värvidega. Igaüks neist muudab tegelasele heidetavat keskkonnavalgust. Kui seniit oleks tumedam (nagu ta seda reaalses maailmas enamasti on), oleksid mudeli ülemised osad tumedamad kui alumised Taeva ja horisondi värvi kasutamine tegeleaste valgustamise mõjutajana sulatab tegelase tükki maad realistlikumalt oma keskkonda ning seetõttu on tulemused fotorealistlikumad.

Sama tulemuse saavutamiseks Blenderis võid kasutada kaudvarju (*Ambient Occlusion*) või valgustada oma tegelast poolkeravalgustitega, mille värvus on sama nagu sinu taeva seniit ja horisont.

Renderdusmootor *Cycles*

Renderdusmootor *Cycles* on uus ja lisandus Blenderi versiooniga 2.61. See on ikka veel arendamisel ning selle eesmärgiks on luua renderdusmootor, mis on interaktiivne ja lihtne kasutada, kuid pakub siiski kõrgtasemel funktsionaalsust. Arendusdokumentatsioon asub [siin](#).

Alustamine

Cycles on Blenderi lisa, mis on vaikimisi aktiveeritud. Selle kasutamiseks tuleb ülemises päiseribas määrata aktiivseks renderdusmootoriks *Cycles*. Kui see tehtud, saab interaktiivse renderdamise sisse lülitada, määrates 3D-vaate joonistusresžiimiks renderdatud pildi (*Rendered*). Renderdus värskendub ka objektide ja materjalide muutmise ajal.

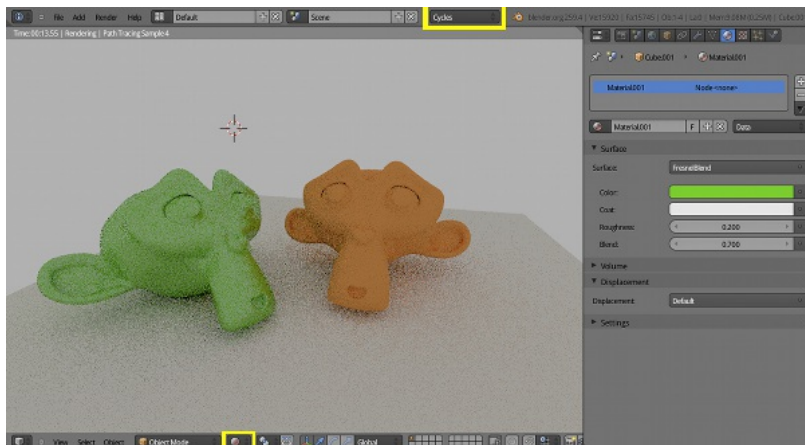
Õppetükid

Lisainfo

- [Kaamera](#)
- [Materjalid](#)
 - [Pind](#)
 - [Maht](#)
 - [Nihkelaotus](#)
- [Maailm](#)
- [Valgustid](#)
- [Sõlmed](#)
 - [Varjutajad](#)
 - [Tekstuurid](#)
 - [Lisasõlmed](#)
- [Valguskiirte teekonnad](#)
- [Integreerija](#)
- [Tekstuuride muutmine](#)
- [Graafikaprotsessoriga renderdamine](#)

Alustamine

Cycles on Blenderi lisa, mis on vaikimisi aktiveeritud. Selle kasutamiseks tuleb ülemises päiseribas määrata aktiivseks renderdusmootoriks *Cycles*. Kui see tehtud, saab interaktiivse renderdamise sisse lülitada, määrates 3D-vaate joonistusresžiimiks renderdatud pildi (*Rendered*). Renderdus värskendub ka objektide ja materjalide muutmise ajal.



Veendumaks, kas ja kuidas sa oma graafikakaarti renderdamiseks kasutada saad, loe [graafikaprotsessoriga renderdamise dokumentatsiooni](#).

Õppetükid

- [Sissejuhatus mootoris Cycles \(Blender Guru\)](#)



- [Sissejuhatus renderusmootoris Cycles \(Blender Cookie\)](#)



- [Kuidas luua mootoris Cycles härmatist \(BlenderDiplom\)](#)



- [Loo realitslik vee simulatsioon \(Blender Guru\)](#)



- [Valgustuse, materjalide ning tekstuuride nipid ja nõksud mootoris Cycles \(BlenderDiplom\)](#)



- [Keerulisemate orgaaniliste materjalide tekitamine sõlmekaartidega \(BlenderDiplom\)](#)



- [Puitvaatidega stseeni renderdamine \(Blender Cookie\)](#)



- [Nutrite ja poltide renderdamine mootoriga Cycles \(Free 3D Tutorials\)](#)



- [Loo realistlik linn \(Blender Cycles\)](#)

[300px](#)

- [Loo protseduuriline tähistaevas \(Blender Cycles\)](#)

[300px](#)

- [Mootori Cycles materjalisõlmed](#)
- [Mootori Cycles õppetükkide seeria](#)
- [Täielik ülevaade mootorist Cycles](#)

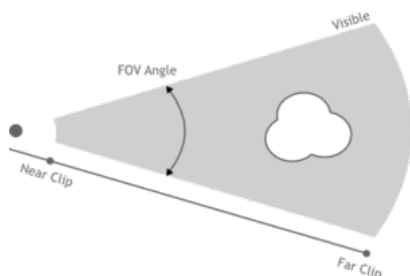
Kaamera

Kaamera ühildub täielikult Blenderi olemasoleva kaameraga, kuid on üks erand: panoraamrenderduse tuge ei ole.

Perspektiiv (*Perspective*)

Lens Size and Angle (objektiivi suurus ja nurk)

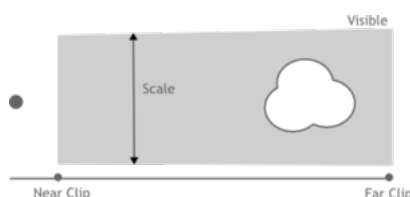
Muudab vaatenurka.



Paralleelprojektsioon(*Orthographic*)

Scale (mõõtkava)

Määrab pildile projitseeritavate objektide suuruse.



Sügavusteravus (*Depth of Field*)

Aperture Size (ava suurus)

Kaamera ava suurus, mille kaudu siseneb valgus (seda kutsutakse mõnikord ka läätse raadiuseks). Kui see on null, on kõik objektid fookuses. Suuremad väärtused muudavad fookuspunkti kaugemale jäävad objektid uduseks. Reaalsete kaamerate puhul võib ava suurust määratleda fookuskauguse ja avaarvu jaotise (*f-stop*) kaudu: $\text{ava suurus} = \text{fookuskaugus} / (2 f\text{-stop})$.

Aperture Blades (objektiivi lamellid)

Kui see säte on 3 või suurem, kasutatakse ümmarguse objektiivi asemel hulknurkset, mis mõjutab renderdatud pildil fookusest väljas olevate heleduslaikude kuju.

Aperture Rotation (objektiivi pööre)

Objektiivi lamellide (Aperture Blades) pööre.

Focal Distance (fookuskaugus)

Kaugus, mille peal objektid on täiuslikult fookuses. Teise võimalusena saab määrata objekti, mille kaugust kaamerast kasutatakse fookuskaugusena.

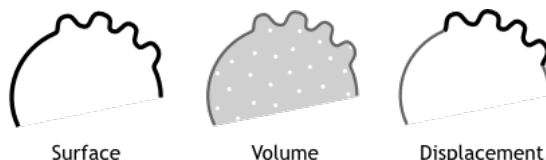
Vaate piiramine (*Clipping*)

Clip Start and End (piirangu algus ja lõpp)

Vahemik, milles asetsevad objektid on otseselt nähtavad. Kõik objektid, mis jäävad selles vahemaast väljapoole, mõjutavad kaudselt siiski pilti, sest välja jäävaid valguse põrkeid ei piirata. OpenGL renderdamise puhul on oluline määrata piirangud väikeste väärtustega, et säiliks piisav rasterdamise täpsus. Kiirtejälitus selle probleemi all eriti ei kannata ning seetõttu saab määrata oluliselt suuremaid väärtusi.

Materjalid

Materjalid määravad võrede, kõverate ja muude objektide välimuse. Need koosnevad kolmest varjutajast (*shaders*), mis määravad vastavalt võre pinna (*surface*), võre sees oleva mahu (*volume*) ja võre pinna nihkelaotuse (*displacement*).



[Pinnavarjutaja \(Surface Shader\)](#)

Pinnavarjutaja määrab valguse käitumise võre pinnal. Üks või enam pinna kahesuunalist hajutamise funktsiooni (BSDF) määravad selle, kas valgus peegeldatakse tagasi, murdub refraktsiooni põhimõttel võresse või neeldub.

Kiirgus (*emission*) määrab selle, kuidas valgust pinnast kiiratakse, ning selle abil on võimalik muuta ükskõik milline pind valgusallikaks.

[Mahu varjutaja \(Volume Shader\)](#)

Mahu varjutaja on antud hetkel arendamisel.

Kui pinnavarjutaja ei peegelda ega neela valgust, siseneb see mahtu. Kui mahu varjutajat pole määratud, liigub valgus otse võrest läbi teisele poolele.

Kui see on määratud, kirjeldab mahu varjutaja valguse käitumist võre mahu läbimisel. Valgus võib ükskõik millises mahu punktis hajuda (*scatter*), neelduda (*absorb*) või kiirguda (*emit*).

Materjalil võib olla korraga nii pinna- kui ka mahu varjutaja või üks kahest. Mõlema kasutamine võib olla kasulik klaasi, vee või jääga sarnanevate materjalide puhul, mille puhul soovid, et osa valgusest pinna läbimisel neelduks ja samal ajal näiteks klaasi pinnal oleva läikevarjutajaga liituks.

[Nihkelaotus \(Displacement\)](#)

Pinna kuju ja selle sisemahtu võivad mõjutada selle nihkevarjutajad. Sel moel saab kasutada tekstuure, mis muudavad võre pinna detailsemaks.

Sõltuvalt sätetest võib nihkekaart olla virtuaalne (mõjutades ainult pinnanormaale, et luua mulje faktuurist – faktuurikaart ehk *Bump map*) või kombinatsioon reaalsest ja virtuaalsest nihutamisest.

Energia säilitamine

Materjalisüsteem on ehitatud füüsikal põhinevat renderdamist silmas pidades. Selgelt on eraldatud see, milline materjal välja näeb, ning see, millist algoritmi selle renderdamiseks kasutatakse. See muudab realistlike tulemuste saavutamise ja tasakaalustatud valgustamise lihtsamaks, kuigi seejuures tuleb paari asja silmas pidada.

Selleks, et materjalid globaalses valgustuses hästi töötaksid, peaksid nad (füüsika terminites) austama energia jäävust. See tähendab, et nad ei tohi peegeldada rohkem valgust, kui sisse tuleb. Selle omaduse täitmist ei valvata, ent kui värvid on vahemikus 0.0 kuni 1.0 ning kahesuunalise hajutamise funktsioonid (BSDF) segatakse kokku ainult varjutaja segamisesõlme *Mix Shader* kasutades, siis vastab energiasäästlikus automaatselt tõele.

Seda on võimalik siiski murda, kui värvide väärtused on suuremad kui 1.0 või kui kasutatakse varjutaja liitmissõlme *Add Shader*, ent nii käitudes tuleb tähelepanelikult jälgida, et materjalid käituksid erinevates valgustingimustes ennustataval moel. Selle tulemusel võib peegeldus tga pörke korral süsteemi valgust juurde lisada, muutudes teatud sorti BSDF-tüüpi kiirgajaks.

Nihkelaotus (*Displacement*)

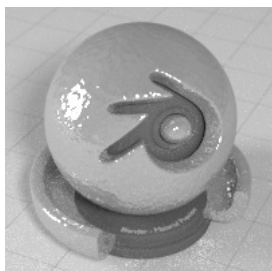
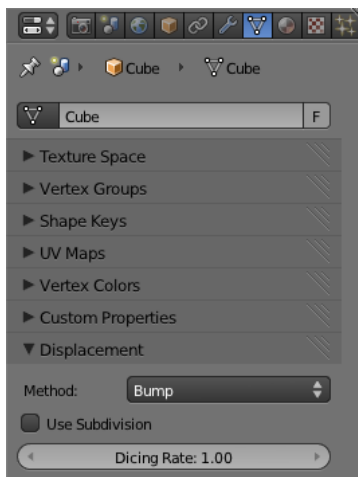
Implementeerimine ei ole veel lõpetatud ning seetõttu määratletakse seda [eksperimentaalsena](#).

Pinna kuju ja võre sees olevat mahtu saab nihkevarjutajatega (*displacement shaders*) mõjutada. Sel moel saab kasutada tekstuure, mis muudavad võre pinna detailsemaks.

Tüüp (*Type*)

Sõltuvalt sätetest võib nihkekaart olla virtuaalne (mõjutades ainult pinnanormaale, et luua mulje faktuurist – faktuurikaart ehk *Bump map*) või kombinatsioon reaalsest ja virtuaalsest nihutamisest. Nihkekaardi tüübi valikuteks on:

- **Tõeline nihutamine (*True Displacement*):** Võre tippe nihutatakse enne renderdamist ja muudetakse seeläbi võret päriselt. Kui võre on korralikult tükeldatud (*subdivided*), annab see valik parima kvaliteediga tulemuse. Seetõttu on see meetod ka kõige mälunõudlikum.
- **Faktuurikaart (*Bump Mapping*):** Kui pinnavarjutajat arvutatakse, kasutatakse tegeliku pinnanormaali asemel muudetud normaali. See on kiire alternatiiv tõelisele nihutamisele, kuid on sisimas ainult ligilähedane tuletis. Pinna siluetid ei ole täpsed ning nihked ei heida varje.
- **Nihe + faktuur (*Displacement + Bump*):** Kahte võimalust saab kombineerida, et kasutada tõelist nihutamist esmaseks võre moonutamiseks ning seejärel faktuurikaarti lõplike detailide lisamiseks.



Faktuurikaardiga nihutamine

Tükeldamine (*Subdivision*)

Implementeerimine ei ole veel lõpetatud ning seetõttu määratletakse seda [eksperimentaalsena](#).

Detailsema nihutamise jaoks tuleb võre tükeldada väikesteks hulknurkadeks. Seda saab teha, lisades pinnatükeldaja töötleja (*Subdivision Surface modifier*), kuid on võimalik lasta ka renderusmootoril endal võret tükeldada.

Pind (*Surface*)

Pinnavarjutaja määrab valguse käitumise võre pinnal. Üks või enam pinna kahesuunalist hajutamise funktsiooni (BSDF) määravad selle, kas valgus peegeldatakse tagasi, murdub refraktsiooni põhimõttel võresse või neeldub.

Kiirgus (*emission*) määrab selle, kuidas valgust pinnast kiiratakse, ning selle abil on võimalik muuta ükskõik milline pind valgusallikaks.

Terminoloogia

- **BSDF** tähistab kahesuunalist hajutamise funktsiooni (*bidirectional scattering distribution function*). See määrab, kuidas valgust pinnal peegeldatakse ja murtakse (refraktsioon).
- **Peegeldumise (*Reflection*)** BSDFid peegeldavad sisse tulevaid kiiri pinna samale küljele.
- **Läbivuse (*Transmission*)** BSDFid kannavad sissetuleva kiire läbi pinna, lastes sellel teiselt poolt väljuda.
- **Murdumise ehk refraktsiooni (*Refraction*)** BSDFid on teatud tüüpi **läbivused**, mis kannavad kiire edasi ja muudavad selle suunda, kui see pinna teiselt poolelt väljub.

BSDFi parameetrid

Peamine erinevus füüsikal mitte baseeruvate renderdajatega on see, et valgustite otsesest valgusest ja kaudsest valgusest tulenevaid peegeldusi ei eraldata, vaid nendega tegeleb üks ja sama BSDF. See vähendab natuke võimalusi, kuid me usume, et lõppkokkuvõttes on see abiks püsivalt sama välimusega renderduste loomisel, mille puhul tuleb määrata vähem parameetreid.

Läikivate BSDF'ide puhul määrab **kareduse (*Roughness*)** parameetrid peegelduse teravuse jaoks – 0.0 on täiuslikult terav ja 1.0 väga hägune. Võrreldes **tugevuse (*Hardness*)** ja **eksponendi (*Exponent*)** parameetritega, on selle eeliseks vahemik 0.0...1.0, mistõttu on selle määramine lineaarsem ja tekstuurimine lihtsam. Suhe on umbes: $karedus = 1 - 1/tugevus$

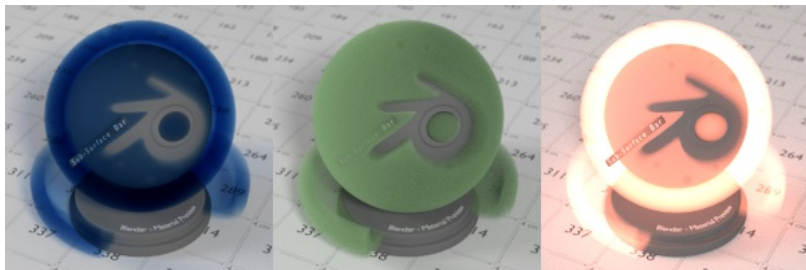
Maht (*Volume*)

Mahu varjutaja on antud hetkel arendamisel.

Kui pinnavarjutaja ei peegelda ega neela valgust, siseneb see mahtu. Kui mahu varjutajat pole määratud, liigub valgus otse võrest läbi teisele poolele.

Kui see on määratud, kirjeldab mahu varjutaja valguse käitumist võre mahu läbimisel. Valgus võib ükskõik millises mahu punktis hajuda (*scatter*), neelduda (*absorb*) või kiirguda (*emit*).

Materjalil võib olla korraga nii pinna- kui ka mahu varjutaja või üks kahest. Mõlema kasutamine võib olla kasulik klaasi, vee või jääga sarnanevate materjalide puhul, mille puhul soovid, et osa valgusest pinna läbimisel neelduks ja samal ajal näiteks klaasi pinnal oleva läikevarjutajaga liituks.



Mahu varjutaja

Tekstuuri muutmise

3D-vaateakna joonistusrežiimid, UV-kaardid ja tekstuuride maalimine toimuvad mootoris *Cycles* sisse lülitamisel natuke erinevalt. UV-kaartidele ei määrata pilditeksteure iseenesest, vaid selle asemel tuleb neile need pildi tekstuuri sõlme *Image* abil määrata.

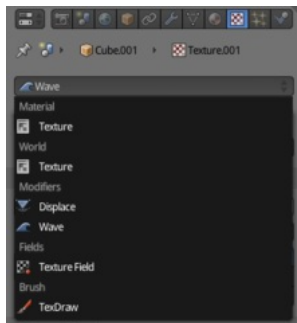
3D-vaateakna joonistusrežiimid

Tekstuuri joonistusrežiim *Texture*, mis toimis Blenderi sisese renderdusmootori kasutamisel, on asendatud kolme teise režiimiga:

- **Tekstuur (*Texture*)**: seda joonistusrežiimi kasutatakse üksikute tekstuuride muutmiseks, maalimiseks ja laotamiseks. Valgustus on sama, mis tahke režiimi (*Solid*) puhul, nii et see režiim on sama kui praegune tekstuuritud tahke (*Textured solid*). Kuvatakse materjali aktiivse pildi tekstuuri sõlme *Image* tekstuuri.
- **Materjal (*Material*)**: GLSL varjutajate abil kuvatakse lihtsustatud versiooni tervest materjalist. See kasutab tahket valgustamist ning on samuti peamiselt kasulik tekstuuride muutmiseks, maalimiseks ja laotamiseks, nähes samal ajal seda, kuidas nad materjaliga suhestuvad.
- **Renderdatud (*Rendered*)**: selles joonistusrežiimis kuvatakse pilt vastavalt valitud renderdusmootorile. Nii autonoomsed (*offline*) renderdajad kui ka mängumootorid saavad selles režiimis vaateakent täieliku valgustusega kuvada.



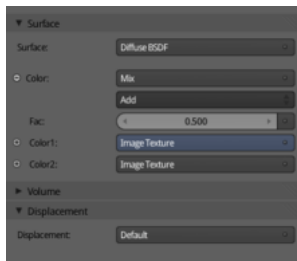
Tekstuuri omadused



Tekstuuri omaduste all saab nüüd tekstuuri valida nimekirjast, milles on välja toodud kõik maailma, valgustite ja materjalide tekstuurisõlmed (kuid samuti näiteks teisendajate, pintslite ja füüsikaväljade tekstuurid).

Varjutussõlmede puhul kasutatakse mootori *Cycles* teksteure. Muudel puhkudel kasutatakse endiselt Blenderi teksteure, kuid tulevikus see muutub.

Maalimine ja UVde muutmise



Tekstuurimaalimise režiimis võetakse pilt, millele maalitakse, aktiivse pildi tekstuuri sõlme *Image*. Selle saab valida kas sõlmedaktorist või tekstuuri omaduste alt ning see on materjali omaduste all sinisega esile tõstetud.

UV-laotuse puhul kasutatakse võre omaduste all määratud aktiivset UV-kaarti. Piltide määramine pildiredaktoris mõjutab samuti aktiivset pildi tekstuuri sõlme.

Maailm (*World*)

Maaailma keskkond saab kiirata valgust, mis põhineb ühel tahkel värvil, füüsikalistel taeva mudelitel või suvalistel tekstuuridel.

Pinnavarjutaja (*Surface Shader*)

Pinnavarjutaja määrab keskkonnast stseeni kiirgatava valguse. Maaailma pind renderdatakse stseenist väga kaugena ning seetõttu ei toimu stseenis olevate objektide ja keskkonna vahel mitte mingisugust kahepoolset suhtlemist, ainult valgus tuleb sisse. Ainus varjutaja, mida toetatakse, on tausta sõlm *Background*, millel on värvi sisend ja jõu faktor, mis määrab valguse intensiivsuse.

Mahu varjutaja (*Volume Shader*)

Iseseisval arendamisel

Nipid

Mõnikord on kasulik kasutada eraldi tausta, mis on otseselt nähtav, selle asemel, et lihtsalt kaudselt objekte valgustada. Lihtne lahendus on lisada segamissõlm *Mix*, mille segamise faktoriks *Blend Factor* on määratud kaamera kiire olemasolu *Is Camera Ray*. Sellisel juhul on esimeseks sisendi värviks kaudne värv ja teiseks otseselt nähtav värv.

Tausta varjutajaid ei sãmplita praegu tähtsuse põhiselt (*importance sampled*), mis tähendab, et nad ei kasuta suuremat hulka sãmpleid kohtades, kus taust on heledam. See võib omakorda viia selleni, et valguse koondumine tekstuuride puhul, millel on üksikud heledad laigud, on aeglane. Taoliste tekstuuride udusemaks muutmine võib müra vähendada ja koondumise kiiremaks muuta.

Valgustid

Lisaks taustast lähtuval valgusele ja objektidele, mis kiirgavad valgust, on stseeni valgustamiseks võimalik kasutada ka valgusteid. Erinevus on selles, et need ei ole renderdatud pildil ise otseselt nähtavad ning neid saab vastavalt oma objektitüübile lihtsamini hallata.

Type (tüüp)

Praegu toetatakse **punkt-** (*Point*), **pind-** (*Area*) ja **päikesevalgust** (*Sun*). Koht- (*Spot*) ja poolkeravalguseid (*Hemi*) ei toetata ning need renderdatakse vastavalt punkt- ja päikesevalgusena, kuid nad võivad tulevikus tööle hakata ning seetõttu on targem neid ühilduvuse huvides mitte sisse lülitada.

Size (suurus)

Lambi suurus Blenderi ühikutes, mille suurendamisel on varjud ja varjutus pehmemad.

Cast Shadow (heida varju)

Selle valiku väljalülitamisel ei blokeeri teele ette jäävad objektid valgustitest tulevate kiirte teekonda. See võib renderdamist kiirendada, sest kiiri ei pea valgusallikani tagasi jälitama.

Punktvalgus (*Point Lamp*)

Punktvalgused kiirgavad valgust igasse suunda võrdselt. Määraates suuruse (Size) nullist suuremaks, muutuvad need kerakujulisteks valgustiteks, millest tingitud varjud ja varjutus on pehmemad.

Pindvalgus (*Area Lamp*)

Pindvalgustid kiirgavad valgust ruudu või ristküliku kujulisest alast ning selle hajumine toimub vastavalt Lambert'i seadusele.

Päikesevalgus (*Sun Lamp*)

Päikesevalgusti kiirgab valgust määratud suunas. Nende asukohta ei arvestata, need on alati väljaspool stseeni lõputult kaugel eemal ning nende valgus ei vähene kaugusest sõltuvalt.

Kuna nad ei asu stseenis, kasutatakse nende tugevuse määramiseks teisi ühikuid ning suuruse väärtus peaks tavaliselt olema väiksem kui teistel valgustitel.

Sõlmed

Materjale, valgusteid ja taustu määratakse kõiki varjutussõlmede võrgustiku abil. Need sõlmed väljastavad väärtusi (*values*), vektoreid (*vectors*), värve (*colors*) ja varjutajaid (*shaders*).

Varjutajad (*Shaders*)

Sõlmekaartide ehitamisel on oluline aru saada mõistest **varjutaja pesa** (*shader socket*). Kõigi pinna- ja mahu varjutajate väljundiks on varjutaja, mis ei kirjelda mitte pinna värvi, vaid seda, kuidas pind või maht valgust mõjutab.

Sõlmedega saab kasutada mitut erinevat tüüpi varjutajat:

- **Kahesuunalist hajutamise funktsiooni** (*BSDF*) varjutaja kirjeldab objekti pinnal toimuva valguse peegeldamist, murdumist ja neeldumist.
- **Kiirguse** (*Emission*) varjutaja kirjeldab valguse kiirgamist objekti pinnal või mahus.
- **Mahu** (*Volume*) varjutaja kirjeldab valguse hajumist mahu sees.
- **Tausta** (*Background*) varjutaja kirjeldab keskkonnast lähtuvat valgust

Igal varjutaja sõlmel on värvi sisend ja varjutaja väljund. Neid saab omavahel varjutajate segamissõlme *Mix Shader* ja varjutajate liitmissõlme *Add Shader* abil segada ja liita. Teised tegevused on keelatud. Tulemuseks olevat väljundit saab seejärel kasutada renderdusmootor, mis arvutab kõik valguste omavahelised suhted kasutamaks otsest ja globaalset valgustust.

Tekstuurid (*Textures*)

Iga tekstuuritüüp on mootori *Cycles* all üks sõlm, millel on sisendiks teksturi koordinaadid ja erinevad parameetrid ning väljundiks värv või väärtus. Tekstuuri andmeblokke (*datablocks*) pole vaja. Selle asemel saab tekstuuride taaskasutamiseks sõlmegruppe kasutada.

UV-laotuse ja vaateaknas tekstuuride maalimise jaoks tuleb kasutada pildi sõlme *Image*. Sellise sõlme aktiivseks muutmisel kuvatakse seda tekstuuriga joonistusrežiimis *Textured* ning sellele saab tekstuuri maalimise režiimis maalida.

Kõigi sõlmede puhul on vaikimisi tekstuuri koordinaatideks genereeritud koordinaadid (*Generated*). Selle erandiks on pildi tekstuurid *Image*, mis kasutavad vaikimisi UV-koordinaate. Igas sõlmes on olemas mõned sätted, mille abil saab muuta tekstuuri laotust ja tulemuseks olevat värvi ning neid saab tekstuuri omaduste alt muuta.

Lisasõlmed

Sõlmed, mis on mõeldud geomeetriliste andmete, tekstuuri koordinaatide, varjutajate kihiti asetamise ning füüsikal mitte põhinevate trikkide jaoks.

Varjutussõlmed (*Shader Nodes*)

Kahesuunalist hajutamise funktsioon (*BSDF*)

Hajuspeegeldus (*Diffuse*)

Lamberti seadusel ja Oren-Nayaril põhinev hajuspeegeldus.

Color input (värvi sisend)

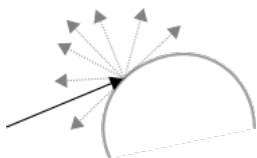
Pinna värv ehk füüsikaliselt öeldes tõenäosus, et igal lainepikkusel peegeldatakse või kantakse üle valgust.

Roughness (karedus)

Pinna karedus. Väärtus 0.0 annab tavalise Lamberti peegelduse, kõrgemad väärtused aktiveerivad Oren-Nayari BSDFi.

BSDF output (kahesuunalist hajutamise funktsioon väljund)

Hajus BSDF varjutaja.



Hajusvarjutaja



Oren Nayar varjutaja
(karedus *Roughness* 1.0)

Läbikumav (*Translucent*)

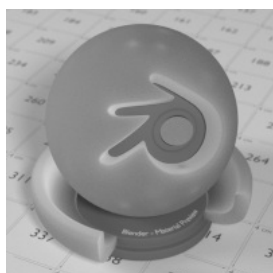
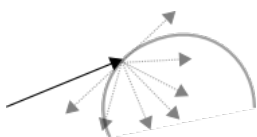
Lamberti seadusel põhinev läbiv hajutamine.

Color input (värvi sisend)

Pinna värv ehk füüsikaliselt öeldes tõenäosus, et mingil lainepikkusel valgus pinda läbib.

BSDF output (kahesuunalise hajutamise funktsioon väljund)

Läbikumav BSDF varjutaja.



Läbikumav varjutaja

Läikepeegeldus (*Glossy*)

Läikiv peegeldus, millel on mikrotahkudega (*microfacet*) jaotus. Seda kasutatakse metalli- ja peeglisarnaste materjalide jaoks.

Läikepeegelduse tulemuseks on tavaliselt mürana palju heledustäppe.

Distribution (jaotus)

Kasutatav mikrokülgede jaotuse tüüp. Sharp (terav) tekitab täiuslikult teravad peegeledused, nagu peegel. Beckmann ja GGX saavad kasutada kareduse sisendit Roughness, et saavutada uduseid peegeldusi.

Color input (värvi sisend)

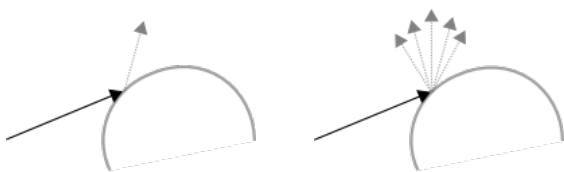
Pinna värv ehk füüsikaliselt öeldes tõenäosus, et mingil lainepikkusel valgust peegeldatakse.

Roughness input (kareduse sisend)

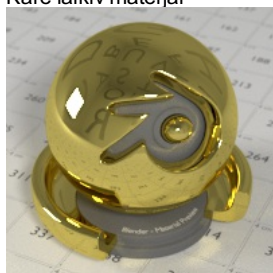
Mõjutab peegelduse teravust. Täiuslikult terav on väärtus 0.0 ning kõrgemad väärtused on sujuvamad.

BSDF output (kahesuunalist hajutamise funktsioon väljund)

Läikiv BSDF varjutaja.



Kare läikiv materjal



Terava peegeldusega läikiv materjal

Läbipaistev (Transparent)

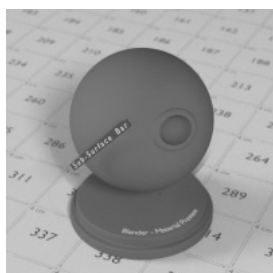
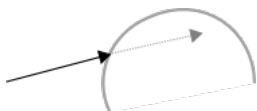
Läbipaistev kahesuunalise hajutamise funktsioon (BSDF), millel puudub refraktsioon (murdumine) ning mille puhul kiired liiguvad otse läbi pinna, justkui geometriat poleks olemas. Kasuta näiteks alfat laotavate kaartidega. See varjutaja [mõjutab valguse teekondi natuke teistmoodi](#) kui teised BSDFid. Pane tähele, et täiesti valged läbipaistvad varjutajad on täiesti läbipaistvad.

Color input (värvi sisend)

Pinna värv ehk füüsikaliselt öeldes tõenäosus, et igal lainepikkusel valgus blokeeritakse või lastakse otse läbi pinna.

BSDF output (kahesuunalist hajutamise funktsioon väljund)

Läbipaistev BSDF varjutaja.



Läbipaistev varjutaja



Tumedam värv muudab läbipaistva varjutaja vähem läbipaistvaks.

Klaas (Glass)

Klaasisamane varjutaja, mis segab nurga alt vaadates valguse refraktsiooni ja peegeldust. Nagu läbipaistva varjutaja puhulgi, muudab ainult puhas valge selle läbipaistvaks. Klaasi varjutaja kipub kaustika (*caustics*) tõttu müra tekitama.

Distribution (jaotus)

Kasutatav mikrokülgede jaotuse tüüp. Sharp (terav) tekitab täiuslikult teravad peegeledust, nagu puhas klaas. Beckmann ja GGX saavad kasutada kareduse sisendit Roughness, et saavutada karedapinnalist klaasi.

Color input (värvi sisend)

Pinna värv ehk füüsikaliselt öeldes tõenäosus, et mingil lainepikkusel valgus pinda läbib.

Roughness input (kareduse sisend)

Mõjutab refraktsiooni ehk murdumise teravust. Täiuslikult terav on väärtud 0.0 ning kõrgemad väärtused on sujuvamad.

IOR input (refraktsiooniindeksi sisend)

Refraktsiooni- ehk murdumisindeks (*Index of Refraction*, IOR), mis määrab, kui palju kiir oma suunda muudab. Väärtuse 1.0 puhul lähevad kiired otse läbi justkui oleks tegemist läbipaistva varjutajaga. Kõrgemad väärtused tekitavad rohkem refraktsiooni.

BSDF output (kahesuunalist hajutamise funktsioon väljund)

Klaas BSDF varjutaja.



Karedus muudab klaasi krobeliseks.



Terav klaas

Samet (Velvet)

Sametpeegelduse varjutaja on mõeldud riidesarnaste materjalide jaoks. Antud hetkel tekitab ka sameti varjutaja palju heledustäppidena esinevat müra.

Color input (värvi sisend)

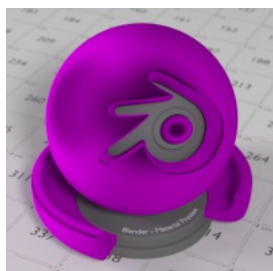
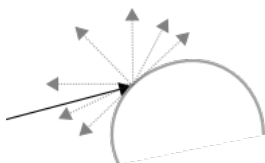
Pinna värv ehk füüsikaliselt öeldes tõenäosus, et mingil lainepikkusel valgust peegeldatakse.

Sigma input (sigma sisend)

kirjeldus puudub

BSDF output (kahesuunalist hajutamise funktsioon väljund)

Samet BSDF varjutaja.



Sametvarjutaja

Kiirgus (*Emission*)

Lamberti seadusel põhinev kiirgus, mida tuleks kasutada materjali ja valgusti pindade väljundina.

Color input (värvi sisend)

Kiirguva valguse värv.

Strength input (tugevuse sisend)

Kiirguva valguse intensiivsus. Punkt- (*Point*) ja pindvalguste (*Area*) puhul on selle ühikuks vatid. Materjalide puhul kindlustab väärtus 1.0 selle, et pildil oleva objekti värv on täpselt sama nagu värvi sisendi kaudu Color saabuv, st materjal muutub 'varjutamatuks' (*shadeless*).

Emission output (kiirguse väljund)

Kiirgav varjutaja.



Kiirgav materjal. Selle heledus on 1 ning seetõttu on see täiuslikult varjutamata.

Taust (*Background*)

Taustavalguse kiirgus. Seda sõlme tuleks kasutada ainult maailma pinna (*world surface*) väljundina. Muudel puhkudel seda ignoreeritakse.

Color input (värvi sisend)

Kiirguva valguse värv.

Strength input (tugevuse sisend)

Kiirguva valguse intensiivsus.

Background output (tausta väljund)

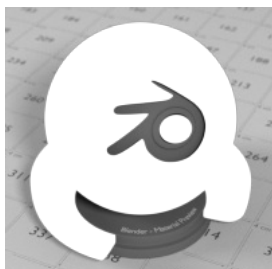
Tausta varjutaja.

Kohahoidja (*Holdout*)

Kohahoidja varjutaja on kasulik komposiitmisel. Sellega luuakse null alfa-läbipaistvusega "auk" kohta, kus selle materjaliga objekt asub.

Holdout output (kohahoidja väljund)

Kohahoidja varjutaja.



Must ala on null alfaga.

Segamine (*Mix*) ja liitmine (*Add*)

Sega või liida varjutajad kokku. Miksimine võib olla kasulik materjalide kihiti asetamise puhul. Sellisel juhul võib näiteks faktori sisend Fac olla ühendatud kaalude segamise sõlmega *Blend Weight*.

Shader inputs (varjutajate sisendid)

Segatavad varjutajad, mida saabuvad kiired tabavad vastavalt faktori pesa Fac väärtuse tõenäosusele.

Fac input (faktori sisend)

Kahe varjutaja segamise kaal (*blend weight*). Nulli puhul kasutatakse ainult esimest ja ühel puhul ainult teist varjutajat.

Shader output (varjutaja väljund)

Segatud varjutaja.



Näiteks haju- ja läikepeegelduste segamise

tulemuseks on kena
keraamiline materjal.

Tekstuurisõlmed

Pildi tekstuur (*Image Texture*)

Kasuta tekstuurina pildifaili.

Image Datablock (pildi andmeblokk)

Pildi allikana kasutatav pildi andmeblokk. Praegu ei toeta *Cycles* kõiki Blenderi poolt toetatavaid pilte. Kõige olulisema puudusena ei toetata praegu genereeritud ja pakitud pilte ning animatsioone.

Color Space (värviruum)

Värviruum, mida sisendpilt kasutab: sRGB või lineaarne (*Linear*). Enamus värvitekstuuride puhul tuleks kasutada vaikimisi sRGBd, kuid näiteks faktuuri- (*bump map*) või alfakaartide (*alpha map*) puhul võib parem olla pikslite väärtuste interpreteerimine lineaarsena (*Linear*).

Vector input (vektorsisend)

Tekstuuri koordinaadid tekstuuri leidmiseks. Kui seda pesa ei ühendata, kasutatakse aktiivse UV-renderduskihi UV-koordinaate.

Color output (värv väljund)

Pildi RGB-värv.

Keskkonna tekstuur (*Environment Texture*)

Kasuta tekstuurina keskkonnalaotuse pildifaili. Keskkonnalaotus peaks olema laiuskraad/pikkuskraad formaadis (*Latitude/Longitude* ehk *lat/long*).

Image Datablock (pildi andmeblokk)

Pildi allikana kasutatav pildi andmeblokk. Praegu ei toeta *Cycles* kõiki Blenderi poolt toetatavaid pilte. Kõige olulisema puudusena ei toetata praegu genereeritud ja pakitud pilte ning animatsioone.

Color Space (värviruum)

Värviruum, mida sisendpilt kasutab: sRGB või lineaarne (*Linear*).

Vector input (vektorsisend)

Tekstuuri koordinaadid tekstuuri leidmiseks. Kui seda pesa ei ühendata, laotatakse pilt keskkonnana niimoodi, et Z-telg on suunatud üles.

Color output (värv väljund)

Pildi RGB-värv.

Taeva tekstuur (*Sky Texture*)

Protseduuriline päike-taevas tekstuur (*sun-sky texture*).

Sun Direction (päikese suund)

Päikese suuna vektor.

Turbidity (hägusus)

Taeva pilvisus.

Vector (vektor)

Tekstuuri koordinaat, milles sellest sãmplit võtta. Kui pesa pole ühendatud, on vaikeväärtuseks genereeritud tekstuurikoordinaadid (*Generated*).

Color output (värv väljund)

Tekstuuri värvi väljund.

Müratekstuur (*Noise Texture*)

Protseduuriline Perlini müra tekstuur, mis on sarnane Blenderi-sisese renderdusmootori pilvetekstuurile *Clouds*

Vector (vektor)

Tekstuuri koordinaat, milles sellest sãmplit võtta. Kui pesa pole ühendatud, on vaikeväärtuseks genereeritud tekstuurikoordinaadid (*Generated*).

Scale input (mõõtkava sisend)

Üldine tekstuuri mõõtkava.

Detail input (detailsuse sisend)

Müra detailsuse hulk.

Distortion input (moonutuse sisend)

Moonutuse hulk.

Color output (värv väljund)

Tekstuuri värvi väljund.

Fac output (faktori väljund)

Tekstuuri intensiivsuse väljund.

Lainetekstuur (*Wave Texture*)

Protseduuriliste vöötide või rõngastega tekstuur, millel on müra moonutus.

Type (tüüp)

Ribade (Bands) või rõngaste (Rings) kujulised lained.

Vector (vektor)

Tekstuuri koordinaat, milles sellest sãmplit võtta. Kui pesa pole ühendatud, on vaikeväärtuseks genereeritud tekstuurikoordinaadid (*Generated*).

Scale input (mõõtkava sisend)

Üldine tekstuuri mõõtkava.

Detail input (detailsuse sisend)

Müra detailsuse hulk.

Distortion input (moonutuse sisend)

Moonutuse hulk.
 Color output (värvi väljund)
 Tekstuuri värvi väljund.
 Fac output (faktori väljund)
 Tekstuuri intensiivsuse väljund.

Voronoi tekstuur (*Voronoi Texture*)

Protseduuriline tekstuur, mis tekitab Voronoi rakke.

Type (tüüp)
 Väljundiks kas intensiivsus (Intensity) või rakud (Cells).
 Vector (vektor)
 Tekstuuri koordinaat, milles sellest samplit võtta. Kui pesa pole ühendatud, on vaikeväärtuseks genereeritud tekstuurikoordinaadid (*Generated*).
 Size input (suuruse sisend)
 Üldine tekstuari mõõtkava.
 Color output (värvi väljund)
 Tekstuuri värvi väljund.
 Fac output (faktori väljund)
 Tekstuuri intensiivsuse väljund.

Musgrave'i tekstuur (*Musgrave Texture*)

Edasiarendatud protseduuriline müratekstuur.

Type (tüüp)
 Multifractal, Ridged Multifractal, Hybrid Multifractal, fBM, Hetero Terrain.
 Vector (vektor)
 Tekstuuri koordinaat, milles sellest samplit võtta. Kui pesa pole ühendatud, on vaikeväärtuseks genereeritud tekstuurikoordinaadid (*Generated*).
 Scale input (mõõtkava sisend)
 Üldine tekstuari mõõtkava.
 Detail input (detailsuse sisend)
 Mõra detailsuse hulk.
 Dimension input (mõõtkava sisend)
Tulemas
 Lacunarity input (lakunaarsuse sisend)
Tulemas
 Offset input (nihke sisend)
Tulemas
 Gain input (tõusu sisend)
Tulemas
 Color output (värvi väljund)
 Tekstuuri värvi väljund.
 Fac output (faktori väljund)
 Tekstuuri intensiivsuse väljund.

Värvülemineku tekstuur (*Gradient Texture*)

Värvülemineku ehk gradiendiga tekstuur.

Type (tüüp)
 Lineaarne (Linear) või kerakujuline (Spherical) värvüleminek.
 Vector (vektor)
 Tekstuuri koordinaat, milles sellest samplit võtta. Kui pesa pole ühendatud, on vaikeväärtuseks genereeritud tekstuurikoordinaadid (*Generated*).
 Color output (värvi väljund)
 Tekstuuri värvi väljund.
 Fac output (faktori väljund)
 Tekstuuri intensiivsuse väljund.

Maagia tekstuur (*Magic Texture*)

Pühkedeelne värvitekstuur.

Depth (sügavus)
 Iteratsioonide arv.
 Vector (vektor)
 Tekstuuri koordinaat, milles sellest samplit võtta. Kui pesa pole ühendatud, on vaikeväärtuseks genereeritud tekstuurikoordinaadid (*Generated*).
 Distortion input (moonutuse sisend)
 Moonutuse hulk.
 Color output (värvi väljund)
 Tekstuuri värvi väljund.
 Fac output (faktori väljund)
 Tekstuuri intensiivsuse väljund.

Ruudustiku tekstuur (*Checker Texture*)

Malelauamustriga tekstuur.

Vector (vektor)

Tekstuuri koordinaat, milles sellest sãmplit võtta. Kui pesa pole ühendatud, on vaikeväärtuseks genereeritud tekstuurikoordinaadid (*Generated*).

Color1/2 input (värv 1/2 sisend)

Ruutude värvid.

Scale input (mõõtkava sisend)

Üldine tekstuuri mõõtkava.

Color output (värvi väljund)

Tekstuuri värvi väljund.

Fac output (faktori väljund)

Tekstuuri intensiivsuse väljund.

Lisasõlmed

Väärtus (Value)

Sisestab arvu (skalaarne väärtus).

Value (väärtus)

Väärtus väljundisse.

RGB

Sisestab RGB-värvi.

Color (värv)

RGB-värv väljundisse.

Geomeetria (Geometry)

Kehtiva varjutuspunkti (*shading point*) geomeetriline informatsioon. Kõik vektori koordinaadid asuvad maailma ruumis (*World Space*). Mahu varjutajate puhul saab kasutada ainult asukohta ja sissetulevat vektorit.

Position (asukoht)

Varjutuspunkti asukoht.

Normal (normaal)

Varjutusnormaal pinnal (arvestab silutud normaale ja faktuurikaarti).

Tangent (puutuja)

Pinna puutuja.

True Normal (tegelik normaal)

Pinna geomeetria ehk tasapinnaline normaal (*flat normal*).

Incoming (sissetulev vektor)

Vektor, mis viitab suunale, millest varjutuspunkti vaadeldakse.

Parametric (parameetrid)

Varjutuspunkti parameetrilised koordinaadid pinnal.

Backfacing (vaade tagant)

1.0, kui külge vaadatakse tagant, 0.0 kui eest.

Tekstuuri koordinaadid (Texture Coordinates)

Tavapäraselt kasutatavad tekstuuri koordinaadid, mida enamasti kasutatakse tekstuurisõlmede vektorsisendis Vector.

Generated (genereeritud)

Võre tippude asukohtades automaatselt genereeritud tekstuurikoordinaadid, mille puhul tippe ei moonutata, sundimaks neid animatsiooni ajal pinna külge hoidma. Vahemikus 0.0 kuni 1.0 moonutamata võre piirdkasti suhtes.

UV

UV-tekstuuri koordinaadid, mis pärinevalt aktiivselt UV-renderduskihiilt.

Object (objekt)

Asukoha koordinaadid objekti ruumis.

Camera (kaamera)

Asukoha koordinaadid kaamera ruumis.

Window (aken)

Varjutuspunkti asukoht ekraanil, mille vahemik 0.0 kuni 1.0 tähistab vastavalt renderduse vasakut ja paremat ning alumist ja ülemist serva.

Reflection (peegeldus)

Teravate peegelduste suunda tähistav vektor, mida kasutatakse enamasti keskkonналаotuste puhul.

Atribuut (Attribute)

Leia objektile või võrele määratud atribuut. Antud hetkel saab sellisel moel leida nende nime järgi UV-kaarte ning tipuvärvi kihte.

Tulevikus lisatakse kihte ja atribuute juurde. Samuti saab sellel moel leida sisemisi atribuute nagu *P* (asukoht ehk *position*), *N* (normaal ehk *normal*), *Ng* (geomeetriline normaal ehk *geometric normal*), kuigi nende leidmiseks on ka lihtsamaid sõlmi.

Name (nimi)

Atribuudi nimi.

Color output (värvi väljund)

Atribuudi abil interpoleeritud RGB-värv.

Vector output (vektor väljund)

Atribuudi abil interpoleeritud XYZ-vektor.

Fac output (faktori väljund)

Atribuudi abil interpoleeritud ühemõõtmeline väärtus.

Laotus (Mapping)=

Teisenda koordinaati. Tavaliselt kasutatakse tekstuuri koordinaatide muutmiseks.

Translation (liigutamine)

Vektori liigutamine.

Rotation (pööre)

Vektori pööramine XYZ-telgedel.

Scale (mõõtkava)

Vektori mõõtkava teisendamine.

Vector input (vektor sisend)

Teisendatav vektor.

Vector output (vektor väljund)
Teisendatud vektor.

Kihtide kaalud (*Layer Weight*)

Väljastab kaalud, mida kasutatakse tavaliselt varjutajate kihindamiseks varjutajate segamissõlme Mix Shader abil.

Blend input (sega sisendid)
Sega esimest ja teist varjutajat.

Fresnel output (fresnel väljund)
Dielektriline Fresneli kaal, mis on näiteks kasulik hajus- ja läikevarjutajate kihiti asetamiseks, et luua plastikust materjali. See toimib nagu Fresneli sõlm Fresnel, kuid selle sõlme sisend on sageli mugavamalt kasutatavamas vahemikus 0.0 kuni 1.0.

Facing output (vaatepunkti väljund)
Kaal, mis sulandab esimesest varjutajat teiseks varjutajaks, kui pind liigub vaataja suunas olekust nurga all olekusse.

Fresnel

Dielektriline Fresnel, mis arvutab seda, palju valgust läbi kihi refraktsiooni alusel murdub ja palju peegeldatakse. Tulemuseks olevat kaalu saab kasutada varjutajate kihindamiseks varjutajate segamissõlme Mix Shader abil. See sõltub pinnanormaali ja vaatepunkti vahelisest nurgast.

IOR input (refraktsiooniindeksi sisend)
Sisestatava materjali refraktsiooni- ehk murdumisindeks (*Index of Refraction*, IOR).
Fresnel output (fresnel väljund)
Fresnel kaal, mis määrab tõenäosuse, et valgus kihilt selle läbimise asemel peegeldub.

Valguse teekond (*Light Path*)

Sõlm, mille abil saad leida, millist tüüpi kiir varjutajat käivitab. See on eriti kasulik füüsikal mittepõhinevate trikkide jaoks. Rohkem informatsiooni iga tüübi tähenduse kohta leiad [Valguse teekondade](#) peatükist.

Is Camera Ray output (on kaamera kiire väljund)
1.0, kui varjutaja käivitab kaamera kiir, vastasel juhul 0.0.
Is Shadow Ray output (on varju kiire väljund)
1.0, kui varjutaja käivitab varju kiir, vastasel juhul 0.0.
Is Diffuse Ray output (on hajunud kiire väljund)
1.0, kui varjutaja käivitab hajunud kiir, vastasel juhul 0.0.
Is Glossy Ray output (on läike kiire väljund)
1.0, kui varjutaja käivitab läikekiir, vastasel juhul 0.0..
Is Singular Ray output (on singulaarne kiir väljund)
1.0, kui varjutaja käivitab singulaarne kiir, vastasel juhul 0.0.
Is Reflection Ray output (on peegelduskiire väljund)
1.0, kui varjutaja käivitab peegelduskiir, vastasel juhul 0.0.
Is Transmission Ray output (on läbiva kiire väljund)
1.0, kui varjutaja käivitab läbivuse kiir, vastasel juhul 0.0.

Valguse teekonnad

Kiirte tüübid

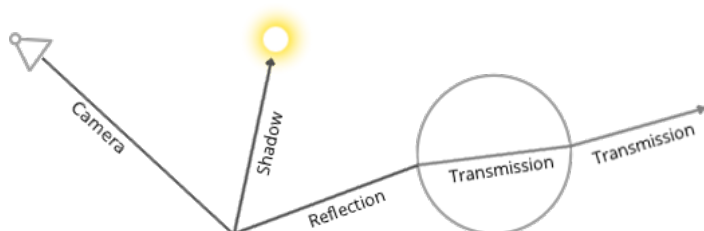
Kiiri saab jagada tüübi järgi nelja kategooriasse:

- Kaamera (*Camera*): kiir tuleb otse kaamerast
- Peegeldus (*Reflection*): kiir tekib pärast mõnelt pinnalt peegeldumist
- Läbivus (*Transmission*): kiir tekib pärast pinna läbimist
- Vari (*Shadow*): kiirt kasutatakse läbipaistvate varjude jaoks

Peegelduse ja läbivuse kiirtel võivad lisaks olla järgnevad omadused:

- Hajusus (*Diffuse*): kiir tekib läbi hajuspeegelduse või läbivuse (läbikumamine ehk *translucency*)
- Läige (*Glossy*): kiir tekib läbi läikepeegelduse või läbivuse
- Singulaarne (*Singular*): kiire tekitab täiuslikult terav peegeldus või läbivus

Kasutada saab valguse teekonna sõlme *Light Path*, et teada saada, millist tüüpi kiire jaoks varjutust arvutatakse.



Pörkek kontroll

Maksimaalset valguse pörgete (*light bounces*) arvu saab käsitsi määrata. Kuigi ideaalis peaks see arv olema lõpmatu, piisab praktikas võib-olla väiksemast pörgete arvust. Mõned valguse vastasmõjud võib meelega jätta arvesse võtmata, et koondumist kiirendada. Ka hajus- ja läikepeegelduste või -läbivuse pörgete arvu saab individuaalselt määrata.

Valguse teekonnad lõpetatakse oletuslikult, kui minimaalne pörgete arv on väiksem kui maksimaalne pörgete arv. Sellisel juhul peatatakse minimaalsest arvust pikemad teekonnad juhuslikult pärast seda, kui nende panus pildi valgustamisse on vähenenud. Selle tulemuseks on küll sama pilt, mis renderdatakse kiiremini, kuid mis võib olla mürarikkam.

Tavapärane müra põhjus on kaustika (*caustics*), mis tähistab hajususe pörkeid, millele järgnevad läikepörked (oletades, et alustasime teekonda kaamerast). On olemas võimalus sellise valguse täielikuks välja lülitamiseks.

Läbipaistvus (*Transparency*)

Pinna kahesuunalise hajutamise funktsiooni (BSDF) varjutajat koheldakse teistest erinevalt. Kui kiir selle varjutaja läbib, liigub valgus otse edasi, justkui poleks seal mitte mingisugust geomeetria. Läbipaistvat BSDF-i varjutajat läbiva kiire tüüp ei muutu.

Alfa väljund on läbipaistvate BSDFide puhul samuti erinev. Teisi läbivuse BSDFe peetakse läbipaistmatuteks, sest nad muudavad valguse suunda. Seetõttu ei saa neid kasutada alfaga komposiitmisel (*Alpha Over*), kuid läbipaistvate BSDFide puhul on see võimalik.

Läbipaistvate pörgete maksimaalset arvu määratakse teistest pörgetest eraldi. Samuti on võimalik tõenäosusel põhinev läbipaistvate pörgete lõpetamine, mis võib aidata kaasa paljudest läbipaistvatest kihtidest koosnevate stseenide renderdamisele.

Pea meeles, et kuigi reaalselt liigub kiir edasi niimoodi, nagu poleks ta vastu geomeetria pörganud, mõjutab iga läbipaistvuse kihi puhul vajalik varjutaja määramine ja kiire jälitamine siiski renderdamise jõudlust.

Kiirte nähtavus

Objektide puhul saab määrata seda, kas nad on järgnevatele kiire tüüpidele nähtavad või mitte:

- Kaamera (*Camera*)
- Hajuspeegeldus (*Diffuse reflection*)
- Läikepeegeldus (*Glossy reflection*)
- Läbivus (*Transmission*)
- Vari (*Shadow*)

Seeläbi saab näiteks valgust kiirgava võre muuta kaamera kiirte jaoks nähtamatuks. Duplitseeritud objektid pärivad originaali nähtavuse – kui ülemobjekt on mõnede kiire tüüpide jaoks peidetud, siis on nende tüüpide jaoks peidetud ka alamobjektid.

Jõudluse seisukohalt on nende sätete kasutamine otstarbekam kui sama tulemuse saavutatav varjutussõlmede kaardi kasutamine. Objektid, mis on teatud tüüpi kiirtele nähtamatud, jäetakse juba kiirte liikumise ajal vahele, mistõttu tuleb heita vähem kiiri ja määrata vähem varjutajaid.

Integreerija (*Integrator*)

Integreerija on renderdusalgorithm, mis arvutab valgustust. *Cycles* toetab praegu ainult otsese valguse sümplite teekonna jälitamise integreerijat. See töötab hästi erinevate valguslahenduste puhul, kuid ei sobi kaustika (*caustics*) ja mõnede teiste keerulisemate valgustamisjuhtude puhul.

Kiiri jälitatakse kaamerast stseeni ning need põrkuvad senikaua ringi, kuni leiavad mõne valgusallika, nagu lambi, valgust kiirgava objekti või maailma tausta. Lampide ja valgust kiirgavate pindade leidmiseks kasutatakse nii kaudset (kiirel lastakse järgida pinna kahesuunalist hajutamise funktsiooni, BSDF ehk *Bidirectional Scattering Distribution Function*) kui ka otsest valguse sümplimist (valitakse valgusallikas ning tõmmatakse selleni kiir).

Stseeni sätted

Sämplid (*Samples*)

Render Samples (renderdussämplid)

Teekondade arv, mida iga lõpliku renderduse piksli kohta järgida. Mida rohkem sümpleid ehk proove võetakse, seda vähema müraga ja täpsem tulemus tuleb.

Preview Samples (eelvaate sümplid)

Vaateakna renderdamiseks kasutatavate sümplite arv.

Seed (seeme)

Juhusliku numbriga genereerija seeme, mille puhul iga erinev väärtus annab erineva müramustri.

Põrked (*Bounces*)

Max Bounces (maksimaalne põrgete arv)

Maksimaalne valguse põrgete arv. Parima tulemuse saamiseks tuleks see väärtus viia maksimumini. Kuid praktikas võib olla kasulik väärtuse vähendamine, et saavutada kiiremat renderdamist. Kui maksimaalseks väärtuseks panna 1, on tulemuseks otsene valgus.

Min Bounces (minimaalne põrgete arv)

Minimaalne põrgete arv, mida iga kiir oma teekonnal teeb – pärast seda kasutab integraator renderdusele kõige vähem infot lisavate kiirte lõpetamiseks vene ruletti. Selle väärtuse suurendamisel tekib vähem müra, kuid renderdusaeg kasvab kõvasti.

Diffuse Bounces (hajususe põrked)

Maksimaalne hajususe põrgete arv.

Glossy Bounces (läike põrked)

Maksimaalne läike põrgete arv.

Transmission Bounces (läbivuse põrked)

Maksimaalne läbivuse põrgete arv.

Läbipaistvus (*Transparency*)

Transparency Max (maksimaalne läbipaistvus)

Maksimaalne läbipaistvuse põrgete arv.

Transparency Min (minimaalne läbipaistvus)

Minimaalne läbipaistvuse põrgete arv, pärast mida kasutatakse elimineerimiseks vene ruletti.

Transparent Shadows (läbipaistvad varjud)

Otsese valguse sümplimiseks kasuta kiirte tee jääivate pindade läbipaistvust, et tekitada varje, mida need läbipaistvad pinnad mõjutavad.

Kaustika (*Caustics*)

No Caustics (keela kaustika)

Kuigi teekondade järgimise meetod toetab piisava arvu sümplite korral põhimõtteliselt kaustika renderdamist, võib see praktikas olla ebaotstarbekas ning tekitada olukorra, kus müra on lihtsalt liiga palju. Selle võimalusega saad kaustika täielikult välja lülitada.

Materjali sätted

Sample as Lamp (sämpel kui valgusti)

Vaikimisi kasutavad valgust kiirgavad objektid nii otsest kui ka kaudset valguse sümplite meetodit, kuid mõnedel puhkudel võib teatud materjalide puhul otseste valguse sümplite keelamine viia müra vähenemiseni. Seda saab teha sättega Sample as Lamp. See on eriti kasulik suurte objektide puhul, mis kiirgavad teiste valgusallikatega võrreldes vähe valgust.

Maailma sätted

Sample as Lamp (sämpel kui valgusti)

Vaikimisi arvutatakse maailmast tulenev valgustus ainult kaudset valguse sümplite meetodile. Kuid keerulisemate keskkonnakaartide puhul võib see olla liiga mürarikas, sest BSDF'i sümplimine ei pruugi lihtsalt leida keskkonnalaotuses pildil olevaid heleduslaike. Selle sätte sisse lülitamisel sümplitakse maailma tausta kui valgustit ning heledamatele osadele antakse automaatselt rohkem sümpleid.

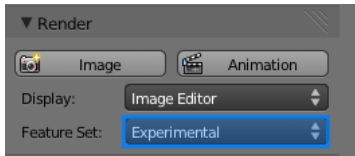
Map Resolution (laotuse resolutsioon)

Kui sämpel kui valgusti sätte *When Sample as Lamp* on sisse lülitatud, siis määrab see sätte olulisuse kaardi suuruse (*importance map*, resolutsioon x resolutsioon). Enne renderdamise algust luuakse olulisuse kaart, mille jaoks eelrenderdatakse maailma varjutaja halltoonides pilt. Seda kasutatakse seejärel määramaks, millised tausta osad on heledad ning peaksid seetõttu kasutama rohkem sümpleid kui tumedad osad. Kõrgem resolutsioon annab täpsema tulemuse, kuid nõuab pikemat eeltöötlust ja rohkem mälu.

Eksperimentaalsed võimalused

Mõned mootori *Cycles* võimalused ei ole veel lõplikud, kuid on juba testimise eesmärgil lisatud. Need võimalused ei pruugi töötada, võivad Blenderi kokku jooksutada ning nende käitumine võib hilisemates versioonides muutuda.

Vaikimisi on nad peidetud, kuid neid saab sisse lülitada, määrares renderduse sätete all funktsionaalsusloendi (Feature Set) eksperimentaalseks (Experimental).



Praegu peetakse eksperimentaalseteks:

- OpenCLi avatud arvutuskeelt kasutavad graafikakaardid (*OpenCL device*)
- Nihkekaarte (*Displacement*)
- Tükeldamist (*Subdivision*)

Graafikaprotsessoriga renderdamine (*GPU Rendering*)

Sissejuhatus

Graafikaprotsessoriga renderdamine (*GPU Rendering*) võimaldab kasutada renderdamisel tavalise protsessori (CPU) asemel graafikakaarti (GPU). See võib renderdamist kiirendada, sest moodsad graafikakaardid on loodud suuremahulise paralleelarvutamise jaoks. Teisalt võib sel olla ka keeruliste stseenide renderdamisel piiranguid (sest mälu on vähem) ning võib tekkida probleem kasutajaliidese reageerimisega, kuna graafikakaart tegeleb samal ajal nii monitorile pildi kuvamise kui ka renderdamisega.

Mootoril Cycles on kaks graafikakaardiga renderdamise režiimi: läbi CUDA, mida peaks NVidia graafikakaardi puhul eelistama, ja läbi OpenCLi, mille eesmärk on toetada renderdamist AMD/ATI graafikakaartidel. Praegusel hetkel on OpenCLi implementatsioon alles eksperimentaalsel tasemel ning võimaldab ainult kaudvarjuga (*ambient occlusion*) samanevate piltide renderdamist.

Seadistamine

Graafikaprotsessoriga renderdamise sisse lülitamiseks mine kasutaja eelistuste akna *User Preferences* süsteemi sakki *System* ning vali sealt soovitud arvutamise seaded (*Compute Device(s)*). Seejärel saad iga stseeni puhul määrata, kas soovid kasutada tavalise või graafikaprotsessoriga renderdamist (renderdusomaduste paneelist *Render*).

CUDA

NVidia CUDA tugi on mõeldud **NVidia graafikakaartidega** renderdamiseks. Me toetame graafikakaarte alates GTX 2xx-st (varjutaja mudel (*shader model*) 1.3), kuid **soovitav on kasutada GTX 4xx või GTX 5xx kaarte** (varjutaja mudel 2.x), sest ainult nendega saavutat sa kiirema tulemuse. Varasemad kaardid on sageli aeglasemad kui tavalise protsessoriga (CPU) renderdamine.

Cycles eeldab kõigil operatsioonisüsteemidel värskete draiverite installimist. Kindlasti laadi endale alla Blenderi versioon, mis vastab sinu operatsioonisüsteemile (64-bitine Blender 64-bitisele operatsioonisüsteemile).

[CUDA kaartide nimekiri koos nende varjutajamudeliga](#)

Vanemad kaardid

Macis ja Linuxis on võimalik kernelit ametlikult mitte toetatud graafikakaartide jaoks jooksvalt kompileerida. GeForce 8xxx, 9xxx kaardid ei ole ametlikult toetatud, kuid need võivad töötada, kui [eksperimentaalsed võimalused](#) sisse lülitada.

Selle jaoks peab olema installeeritud [CUDA tööriistakomplekti versioon 4.0](#) või uuem (64-bitine versioon). Teised versioonid võivad töötada, kuid need ei ole toetatud. Kui renderdamist esimest korda sooritatakse, tuleb renderdustuum (*kernel*) vastavalt sinu graafikakaardi arhitektuurile kompileerida. Kuna Cycles on tavalisest graafikakaardi kernelist suhteliselt keerukam, võib kompileerimine kesta 40 sekundit kuni paar minutit. See võib ka graafikakaardi mudelist sõltuvalt nõuda kuni 2 GB mälu.

OpenCL

Selle implementeerimine ei ole veel lõpetatud ning seetõttu määratletakse seda [eksperimentaalsena](#).

OpenCL toel renderdamine ei ole veel täielik, kuid sellega tegeletakse ning varsti toetame suuremat arvu graafikakaarte. Praegu toetame ainult lihtsalt savi renderdust (*clay rendering*), sest AMD OpenCL kompilaator ei kompileeri meie täielikku tuuma ära.

Kui renderdamist esimest korda sooritatakse, tuleb renderdustuum (*kernel*) vastavalt sinu graafikakaardi arhitektuurile kompileerida. Kuna mootor *Cycles* on tavalisest graafikakaardi tuumast suhteliselt keerukam, võib kompileerimine kesta 40 sekundit kuni paar minutit. See võib ka graafikakaardi mudelist sõltuvalt nõuda kuni 2 GB mälu.

Nõutav on OpenCL versioon 1.1 või uuem.

Korduma kippuvad küsimused

=Miks Blender renderdamise ajal ei vasta?

Sel ajal, kui graafikakaart renderdab, ei saa see kasutajaliidest uuesti kuvada, mistõttu liidesega suhtlemisel tagasiside kehveneb. . Me proovime seda probleemi vältida, andes kontrolli graafikakaardi üle tagasi nii sageli kui võimalik, kuid täiesti sujuvat kasutamist me garanteerida ei saa (eriti väga keeruliste stseenide puhul). See on graafikakaartide piirang, millele ühtegi reaalselt lahendust ei eksisteeri, kuid võib-olla suudame seda tulevikus natuke parandada.

Võimaluse korral tuleks installeerida rohkem kui üks graafikakaart, mille puhul üks kuvaks liidest ja teine/teised renderdaksid.

Miks stseeni, mida tavaline protsessor renderdab, graafikakaardiga ei renderdata?

Sellel võib olla mitmeid põhjusi, kuid kõige sagedamini on põhjuseks see, et su graafikakaardil pole piisavalt mälu. Praegu saame renderdada ainult neid stseene, mis mahuvad graafikakaardi mällu, ning see on tavaliselt väiksem kui protsessori oma. Pea meeles, et näiteks 8k, 4k, 2k ja 1k pilditekstuur võtavad vastavalt 256 MB, 64 MB, 16 MB ja 4 MB mälu.

Me kavatseme lisada toe stseenidele, mis on suuremad kui graafikakaardi mälu, kuid seda ei juhtu lähiajal.

Kas saan renderdamiseks kasutada mitut graafikakaarti?

Seda veel ei toetata, kuid plaanis on see lähitulevikus lisada.

Kas mitu graafikakaarti suurendaksid kasutatava mälu hulka?

Ei, iga graafikakaart saab lugeda ainult iseenda mälu.

Mis renderdab kiiremini – NVidia või AMD, CUDA või OpenCL?

Praegu renderdab CUDAGA NVidia kiiremini. Pole ühtegi reaalselt põhjust, miks see nii peaks olema, sest me ei kasuta ühtegi CUDAlle ainuomast võimalust, kuid selle kompilaator tundub olevat arenenum ning toetab seega paremini suuri arvutustuumasid. OpenCLi tugi on ikka veel arendamisel ning seda pole samapalju optimeeritud, sest terve kernel pole veel korraga tööle hakanud.

Veateated

Unsupported GNU version! gcc 4.5 and up are not supported!

Linuxil võid lähtuvalt oma GNU kompilaatorikogust (*GNU Compiler Collection*, GCC) saada veateate, et toetatakse ainult versioonist 4.5 uuemaid GNU versioone.

Kui selle teate saad, siis kustuta failist `/usr/local/cuda/include/host_config.h` järgnev rida

```
#error -- unsupported GNU version! gcc 4.5 and up are not supported!
```

CUDA Error: Invalid kernel image

Kui saad selle vale tuuma (*kernel*) veateate Windowsi 64-bitises versioonis, siis tee kindlaks, kas sa ikka kasutada Blenderi 64-bitist versiooni ja mitte 32-bitist.

CUDA Error: Out of memory

See mälu otsasaamise veateade tähendab, et graafikakaardil ei ole piisavalt mälu, millesse stseeni salvestada. Praegu saame renderdada ainult neid stseene, mis mahuvad graafikakaardi mällu, ning see on tavaliselt väiksem kui protsessori oma. Loe selle kohta täpsemalt ülaltpoolt.

The NVIDIA OpenGL driver lost connection with the display driver

See veateade ütleb, et NVidia OpenGL draiver kaotas kuvamisdraiveriga ühenduse, kuna ooteaja (*Windows Time-Out*) piir ületati, ning ei saa enam tööd jätkata.

Kui graafikakaarti kasutatakse nii liidese kuvamiseks kui ka renderdamiseks, on Windowsil piirang, kui kaua aega võib graafikakaart renderdamise arvutuste peale kulutada. Kui sul on eriti keeruline stseen, võib Cycles kulutada selle peale liiga palju graafikakaardi aega. Jõudluse paneelis *Performance* oleva renderdustükkide suuruse sätte *Tile Size* väärtuse vähendamine võib seda probleemi parandada, kuid tõeline lahendus oleks kasutada renderdamiseks ja liidese kuvamiseks eraldi graafikakaarte.

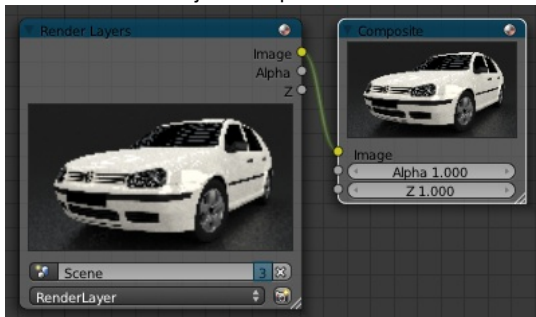
Komposiitmissõlmed

Komposiitmissõlmede (*Compositing Nodes*) abil saad pilti (või videot) üheaegselt nii kokku panna kui ka täiustada. Komposiitmissõlmedega saad kaks videoklippi kokku kleepida ning tervet tulemust korraga toneerida. Saad muuta üksiku pildi või terve videoklipi värve nii staatiliselt moel kui ka dünaamiliselt, nii et need muutuvad aegamööda (kui klipp edasi mängib). Sedasi saad kasutada komposiitmissõlmi nii videoklippide kokku monteerimiseks kui ka nende täiendamiseks.

Termin: Pilt (*Image*)

Me kasutame terminit "pilt", et viidata üksikule pildile, ühele pildile numereeritud piltide jadas või videoklipi ühele kaadriks.

Sõlmekaart arvutab jada ühe pildi haaval hoolimata sellest, millist sisendit sa kasutad.



Piltide muutmiseks kasutad sa sõlmi, et importida pilt Blenderisse, selle muutmiseks, selle teiste piltidega kokku monteerimiseks ning viimaks selle salvestamiseks.

Vasakul olev pilt kujutab kõige lihtsamat sõlme nuudlit - sisendi sõlm *Input* lõimib kaamera vaate väljundi sõlme *Output*, et seda saaks salvestada.

Sõlmede mõisted

Sõlmed (*Nodes*)

Sõlmed (*Nodes*) on individuaalsed blokid, mis viivad läbi mingi kindla operatsiooni ning millel võib olla üks või mitu erinevat väljundit.

Kontseptuaalselt on olemas kolme üldist tüüpi sõlmi:

- Sisendi sõlmed (*Input Nodes*)**

need sõlmed "toodavad" informatsiooni, kuid neil endil puuduvad sisendid.

Näited: renderduskihid (*Render Layers*), väärtus (*Value*) ja *RGB*.

- Muutmissõlmed (*Processing Nodes*):**

need sõlmed "filtreerivad" või "teisendavad" oma sisendeid, et luua üks või mitu väljundit.

Näited: *RGB*-kõverad (*RGB Curves*), sügavusudu (*Defocus*) ja vektorhägu (*Vector Blur*).

- Väljundi sõlmed (*Output Nodes*):**

need sõlmed "neelavad sisendid alla" ja loovad neist mingit sorti mõistusepärase tulemuse.

Näited: komposiit (*Composite*, mis määrab Blenderi poolt kasutatava lõpliku väljundi), eelvaataja (*Viewer*, mis näitab mõne pesa väljundit) ja faili väljund (*File Output*).

Nuudlid (*Noodles*)

Sõlmede peamine mõte seisneb selles, et saad neist luua ükskõik kui keeruka sõlmede "võrgustiku", ühendades ühe või enama sõlme "väljundid" ühe või enama sõlme "sisenditega". Seejärel saad (vastavalt oma soovile) määrata iga sõlme puhul vajalikke parameetreid.

Seda võrgustikku kutsutakse "nuudliks" ning see kirjeldab seda, kuidas informatsioon sõna otseses mõttes sellest "läbi voolab", et tekitada sinu poolt soovitud tulemus.

Sõlmede grupid (*Node Groups*)

Saad luua "sõlmede gruppe" ning kasutada neid samal moel, nagu nad oleksid üksik sõlm.

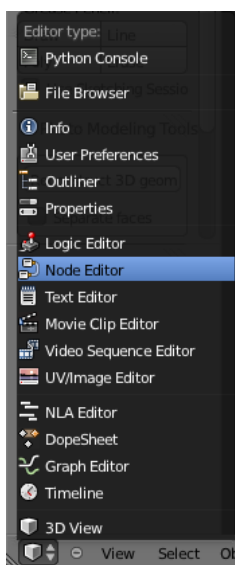
Neid gruppe saab teistest failidest linkida ja lisada.

Sõlmeredaktorisse sisenemine ja sõlmede aktiveerimine

Sisene [sõlmeredaktorisse \(Node Editor\)](#) ja lülita komposiitmissõlmed (Composite Nodes) sisse, vajutades "näo" ikoonile.

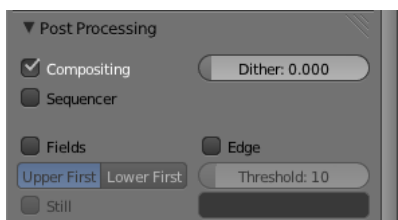


Sõlmeredaktori päis koos sisse lülitatud komposiitmissõlmedega (Composite Nodes)



Sõlmedaktori akna valimine

Komposiitimise eesmärgil sõlmede aktiveerimiseks vajuta sõlmede kasutamise nuppu Use Nodes. Blender loob vaikimisi alustamise nuudli, mis koosneb kahest kokku lõimitud sõlmest.



Kasuta komposiitmissõlmi

Selle minikaardi kasutamiseks pead nüüd Blenderile ütlema, et ta kasutaks äsja loodud komposiitmissõlmede kaarti ning komposiidiks pildi seda kaarti kasutades. Selle saavutamiseks mine stseeni nupu alla Scene (F10) ja aktiveeri animeerimise nupu Animation all asuv **komposiidi tegemise nupp Do Composite**. See käsib Blenderil komposiitida lõplik pilt, jooksutades selle esmalt läbi komposiitmissõlmede kaardi.

Sul on nüüd olemas esimene sõlmeasetus ehk nuudel, mis kujutab renderduskihi sisendsõlme *RenderLayer*, mis on lõimitud komposiidi väljundsõlme *Composite*. Edasi võid ühendada mitmeid [komposiitmissõlmede tüüpe](#), luues vastavalt enda soovile kaardi (või vastavalt füüsilistele mälu piirangutele, kui need varem kätte jõuavad).

Näited

Piltidega saad sõlmesid kasutades teha peaaegu kõike, mida soovid.

Filmitud video, mis kujutab näitlejat sinise ekraani ees, või midagi tegev renderdatud objekt on võimalik kihina tausta peale asetada. Komposiidi mõlemad kokku ja tulemuseks ongi komposiiditud video.

Võid pildi tunnetust muuta:

- Pildi külmemaks muutmiseks lisatakse sellele sinakas toon.
- Meenutuse või mälestuse edasi andmiseks võib pilti pehmeks muuta.
- Viha või frustratsiooni väljendamiseks lisa pildile punakas toon võib tõsta punase intensiivsust. Selle kõige võimsamaks näiteks on film "Sin City".
- Üllatavat sündmust võib teravamaks muuta ja sellele kontrastsust lisada.
- Õnneliku sündmust - täpselt nagu arvasid - annab edasi ere ja päikeseline kollane (võrdsetes osades punast ja rohelist ilma siniseta).
- Pildile lisatakse sageli pilve tekstuurina peale tolmu ja õhusaastet, et sellele realismi lisada.

Sõlmeredaktor (*Node Editor*)

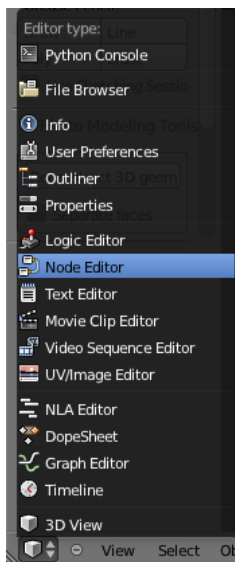
See osa seletab antud akent üldiselt, selle päisemenüüde valikuid ning kuidas sõlmede endiga ringi käia. Samuti seletab see, kuidas Blenderit sõlmi kasutama panna.

Märkus

Lisainformatsiooni selle kohta, **kus sõlmi kasutada**, loe järgmistelt lehekülgedelt:

- [Materjalide sõlmed](#)
 - [Materjalisõlmede tüübid](#)
- [Tekstuuride sõlmed](#)
 - [Tekstuurisõlmede tüübid](#)
- [Komposiitimise sõlmed](#)
 - [Komposiitimissõlmede tüübid](#)

Sisenemine sõlmeredaktorisse



Sõlmeredaktori akna valimine.

Sisene kõigepealt sõlmeredaktorisse, muutes akna tüübi sõlmeredaktori omaks (*Node Editor*). Nagu näha pildilt *Sõlmeredaktori akna valimine*, tuleb vajutada akna tüübi ikoonile ja valida hüpiknimekirjast Node Editor. Kuna sõlmekaardid võivad muutuda üpris suureks, kasuta suurt akent või loo see. Aknal on ruuduline taust ja päis.

Igas .blend-failis olevas stseenis saab olla mitu materjalisõlmede kaarti ja ÜKS komposiitimissõlmede kaart. Sõlmeredaktor näitab sõltuvalt valikust ühte neist kahest kaarditüübist.

Vihje

Soovitame lisada uue aknapaigutuse nimega "Komposiitimine" (aknapaigutuste nimekiri asub ekraani üläääres olevas infopäises), mis koosneb peamiselt ühest suurest sõlmeredaktori aknast. Minu paigutuses on nuppude aken alumises ääres ja märkmete pidamiseks külje peal tekstiredaktori aken. Kui sul on laiekraan (või isegi tavaline), võid soovi korral lisada sõlmede aknapaigutuse vasakusse serva 3D-vaate (3D View) või UV/pildi redaktori akna (UV/Image Editor): nii saad sõlmede muutmise ajal töötada ka piltide või oma mudeli kallal. Materjalisõlmede kohendamise puhul on kasulik hoida objekti kõrval lahti 3D eelrenderduse paneeli.

Nagu näha pildilt, on akna päis esmasel käivitamisel seadistusteta:



Vaikimisi sõlmeredaktori päis.

Sõlmede aktiveerimine

Materjali-, teksturi- ja komposiitimissõlmi tuleb kasutamiseks erinevalt aktiveerida. Aktiveerimise kohta vaata vastavatest peatükkidest.

Sõlmeredaktori akna tegevused

Kui kursor on aknas, saab kasutada mitut tavalist Blenderi kiirvalikut ja hiire liigutust, muuhulgas:

Kustuta

X või Del - kustutab valitud sõlme(d).

Piirdkastiga valimine

B - alustab piirdkastiga valimist. Liiguta kursor õigesse kohta, vajuta LMB ning lohista hiirt sõlmede grupi valimiseks.

Ühenduste läbilõikamine (piirdkastiga)

Vajuta **LMB**  ja lohista - alustab piirdkastiga valimist, KUID hiireklahvi vabastamisel lõigatakse läbi kõik kasti sisse jäävad ühendused.

Ennista

CtrlZ - väga kasulik, kui unustasid enne piirdkastiga valmist vajutada **B**.

Tee uuesti

CtrlY või **Ctrl⇧ShiftZ** - saad seda kasutada, kui "ennistasid" natuke liiga palju :)

Vali mitu

⇧ **Shift LMB**  või ⇧ **Shift RMB**  - vali mitu sõlme.



Haara/liiguta

G - liigutab valitud objekti.

Teosta (Execute)

E - surub sisendid läbi nuudlite (sõlmeasetuste) ja värskendab kõike.

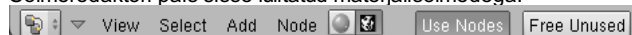
Sõlmeredaktori juhtelemendid

Sõlmede kaardid võivad muutuda üpris karvasteks (ehk siis suurteks ja keerulisteks). Akna sisu (sõlmekaarti) saab liigutada samamoodi nagu iga teise Blenderi akna oma: vajutades **MMB**  ja hiirt ringi lohistades. Hiireratta **Wheel**  üles-alla kerimine numbriklahvistiku nupud + **NumPad**/- **NumPad** suurendavad/vähendavad vaadet. Akna suurust saab muuta ning selle sisu kombineerida, kasutades tavalisi aknakäske (vaata *Kolmemõõtmelises ruumis navigeerimine*).

Sõlmeredaktori päis



Sõlmeredaktori päis sisse lülitatud materjalisõlmedega.



Sõlmeredaktori päis sisse lülitatud komposiitmissõlmedega.

Akna päises näed järgnevaid elemente:

- **View (Vaade)** - näitab asju selgemalt;
- **Select (Vali)** - tee asju selgemalt;
- **Add (Lisa)** - sõlmede lisamiseks, organiseeritud tüübi järgi;
- **Node (Sõlm)** - valitud sõlmedega toimetamiseks;
- **Materjali-, teksturi- ja komposiitmissõlmede vaate valik**;
- **Sõlmede kasutamise nupp Use Nodes**
- **Kasutute sõlmede vabastamise nupp Free Unused**.

Vaate, valimise ja lisamise päisemenüüd

Nende menüüde kaudu saad teha kõige algelisemaid toiminguid:

Vaade (View)

See menüü muudab aknavaadet ning dubleerib tavalisi klaviatuuri kiirvalikuid + **NumPad** (suurenda), - **NumPad** (vähenda), **↵** **Home** (näita kõike) või nendega võrdväärseid hiireliigutusi.

Vali (Select)

Selle menüüga saad valida ühte sõlme või sõlmede gruppi ning see teeb sama, mida **A** vajutamine kõige valimiseks või piirdkastiga valimise alustamiseks **B** vajutamine.

Lisa (Add)

Siit menüüst saad sõlmi lisada. Loe järgmist peatükki, kus kirjeldatakse erinevaid sõlmetüüpe, mida saad lisada, ning seda, mida nad teevad. Sellele menüüvalikule vajutamine on sama, mis kiirvaliku ⇧ **ShiftA** vajutamine, kui kursor on aknas.

Sõlm (Node)

Selles menüüs on sõlmega töötamise tööriistad.

Tööriistad

Näita tsüklilisi sõltuvussuhteid (Show Cyclic Dependencies)

C - olgu, sa oled lisanud ja ühendanud sõlmi täpselt nagu hing ihkab ja arvuti mälu ei ole veel otsa lõppenud. Valides **Show Cyclic Dependencies**, näidatakse sulle, kus oled lõimed ringikujuliselt ühendanud. Näiteks võid lihtsalt ühendada segamise väljundi mõne teise sõlme sisendisse ning seejärel ühendada tolle sõlme väljundi tagasi segamise sisendisse, mille tulemusena on valminud ring, mida mööda piit üha edasi ja edasi jookseb. Kui see ring omaette jätta, väsis see lõpuks ära, hakkab iiveldama ja jooksub su arvuti kokku.

Peida (Hide)

H - peidab valitud sõlmed. Samamoodi nagu tipud võres.

Grupeerimine (Grouping)

Kõige tähtsam, mida selle menüüvalikuga teha saad, on iseenda poolt määratud sõlmede grupi loomine. Grupi saab seejärel

muuta ja kaardile lisada. Grupi loomiseks vali soovitud sõlmed ning kasuta menüükäsku Node → Make Group või lihtsalt klaviatuuri kiirvalikut CtrlG. Nime saad muuta väikses grupis asuvas sisestuskastis. Gruppe on lihtne ära tunda nende rohelisest päisest ja lähedatest nimedest, mida oled neile andnud.

Kustuta (Delete)

X - kustutab valitud sõlmed.

Duplitseeri (Duplicate)

⇧ ShiftD - tekitab linkimata koopja, mille seaded on samad kui originaalil.

Liiguta (Grab)

G - liigutab sõlmi vastavalt hiirele (nagu võrede puhulgi).

Duplitseeri (Duplicate) pettis sind

Uus koopia asetatakse **otse vana peale**. Kuid see ei ole ühendatud ning seetõttu ei mõjuta selle muutmine sinu pilti hoolimata sellest, et see **näeb välja** nagu oleks ühendatud, sest tema **all** olevast sõlmest tulevad välja korralikud lõimed. Et näha õigesti ühendatud sõlme, pead duplitseeritud sõlme kõigepealt eest nihutama.

Liiguta (Grab) - Väike meeldetuletus

See menüükirje ei tegelikult midagi; see on lihtsalt olemas, et tuletada sulle meelde, et võid vajutada nuppu G, kui kursor on aknas ja sa saad ka midagi reaalselt ära teha (näiteks liigutada sõlmi aknas ringi).

Tüübi valimise nupp

Sõlmed on kategooriatesse jaotatud vastavalt sellele, mida nad mõjutavad. Materjalisõlmed mõjutavad .blend-failis kasutatavaid materjale. [Materjalisõlmedega](#) töötamiseks vajuta materjali tähistavale kerale. Kui soovid töötada [komposiitmissõlmedega](#), vajuta komposiitmissõlmede kaardi vaatamiseks fotosid kujutaval nupul.

Sõlmede kasutamise päisenupp (Use Nodes)

See nupp ütleb su renderusmootorile, kas sa soovid lõpliku pildi renderdamisel kasutada materjali värvi arvutamiseks sõlmekaarti või mitte. Kui mitte, siis kaarti ignoreeritakse ning kasutatakse materjali sakkide või stseeni renderdamiseks kõige lihtsamaid seadeid.

Sõlmede lisamine ja paigutamine

Sõlmeredaktorisse lisatakse sõlmi kahel moel:

- Kasutades sõlmeredaktori tööriistaribal olevat lisamise menüüd Add ja valides sealt soovitud sõlmetüübi või
- Hoides kursorit sõlmeredaktori aknas, vajutades kiirvalikut ⇧ ShiftA ning valides hüpikmenüüst Add mõne sõlme.

Üldiselt proovi paigutada sõlmi aknas nii, et pildi arvutamine toimuks vasakult paremale, ülalt alla. Sõlme liigutamiseks vajuta mõnele selle neutraalsele alale ja lohista siis ringi. Sõlme peale võib vajutada peaaegu ükskõik kuhu ning seejärel seda ringi lohistada; ühendused proovivad ennast võimalikult hästi Bezier' kõveratena ümber kohendada.

Sõlmede alla jäävad lõimed

Lõimed (kõverad, mis pesasid ühendavad) võivad liikuda mõne sõlme taha, kuid nad lihtsalt asuvad seal ning ei suhtle tolle sõlmega mitte kuidagi.

Sõlmede alla jäävad sõlmed

"Nüüd sa alles saad!" Kui sa mõnd sõlme duplitseerid, siis asetatakse uus sõlm *täpselt* vana peale. Kui sõlme sinna jätkadki (ja seda võib julgelt teha), siis sa lihtsalt ei näe, et seal on tegelikult *kaks* sõlme! Jah, sõlme vari võib olla natuke tumedam kui enne, kuid enam ei ole see visuaalselt selge, mis on millega ühendatud. Kui tekib kahtlusi, haara sõlmest kinni ja liiguta seda natuke, et näha, kas selle all miski ennast varjab.

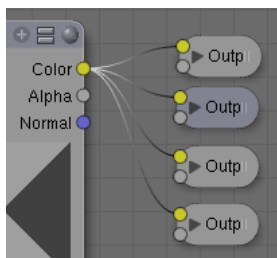
Pesad



Sõlme pesad.

Igal sõlmede aknas asuval sõlmel on "pesad" (sageli nimetatakse neid ka "ühendusteks" või sisendiks-väljundiks), mis kujutavad endast väikeseid värvilisi ringe, millesse ühendatakse sisend- ja väljundandmed (vt pilt *Sõlme pesad*).

Sõlme vasakul küljel olevad pesad tähistavad *sisendeid*, 'paremale jäävad pesad väljundeid'.



Sõlmede ühendamine.

Elu lihtsustamiseks on pesad värviga tähistatud vastavalt sellele, mis tüüpi informatsiooni nad eelistavad saata või vastu võtta. Värv on kolm:

Kollased pesad

Tähistab seda, et pesa võetakse vastu või sellest saadetakse sõlme **värvi** informatsiooni.

Hallid pesad

Tähistab väärtustega (**numbrilist**) informatsiooni. See võib olla kas üksik numbriline väärtus või "väärtuste kaart". (Väärtuste kaarti võid ette kujutada halltoonides kaardina, kus erinev heleduse/tumeduse hulk väljendab iga eri punkti väärtust.) Kui "väärtuste kaardi" pesas kasutada üksikut väärtust, kasutavad kõik kaardi punktid sama väärtust. Tavaline kasutamisiis: alfakaardid ja sõlmede numbrilise väärtusega väljad.

Sinised/lillad pesad


Tähistab **vektor-/koordinaat-/normaalide** informatsiooni.


Kui sa ei kasuta *konverterit*, millest räägime hiljem, tuleb sõlmede vahel kollased pesad ühendada kollastega, hallid hallide ja sinised sinistega.

Värvi kõrval näed sõlmes vastava pesa nime. Kuigi see ei ole alati nii, võib nime pidada vihjeks selle kohta, mis laadi infot pesa eeldab. Kuid see ei tähenda, et info *peab* seda tüüpi olema. Ma võin näiteks ühendada alfata sisaldava pesa nimega (Alpha) halli värvi pesa mõne materjalisõlme halli peegelduvuse (Reflection) pesa ning saavutada ikkagi tulemuse. Kõige olulisem on selle puhul, et tegemist on "hall-hall" ühendusega.

On erandeid, mille puhul saad ilma konverteriteta segada kollast (*nt* värviline pilt) ja halli (*nt* halltoonid). Tavaliselt lisab Blender ise konverteri, kui seda vaja on, seega eksperimenteeri vabalt. Saad kasutada vaataja väljundisõlmi "Viewer", mida seletame hiljem, nägemaks, kas ja kuidas su eksperiment töötab.

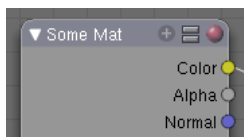
Pesade ühendamine ja lahtiühendamine

Pesade ühendamiseks vajuta soovitud pesale klahviga LMB , hoia seda all ning lohista lõim mõne teise pesani. Sobiva pesani jõudes võid hiire lahti lasta. Samuti saad pesasid ühendada, tähistades neist kaks ning kasutades kiirvalikut F. Sõlmede uuesti ühendamiseks kasuta kiirvalikut CtrlF.

Pesadevahelise ühenduse katkestamiseks hoia all Ctrl LMB  ja lohista hiirt üle ühenduse.

Väljundpesadest võib tõmmata mitu lõime ning ühendada need erinevate sõlmedega (vt pilti *Sõlmede ühendamine*). Sellisel juhul saadetakse mööda igat lõime väljundi koopiat. Sisendpesa saab ühendada aga ainult lõime.

Sõlme juhtelemendid



Sõlme ülaserv.

Sõlme ülaservas on kuni neli sõlme kontrolleri (vt pilti *Sõlme ülaserv*). Nendele vajutamine määrab selle, palju informatsiooni sõlm näitab.

Nool

Vasakul olev nool voldib sõlme täielikult kokku (vt pilti *Voltimise nool*).

Plussmärk (+)

See ikoon lükkab kokku kõik pesad, mis ei ole lõimedega ühendatud (vt pilti *Plussmärk*).

Kaks ruutu (=) või "Võrdusmärk"

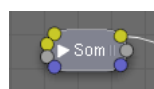
Kahe ruuduga ikoon lükkab kokku kõik sõlmes olevad väljad, mis sisaldavad kaste, milles on info (vt pilti *Menüü kokkuluukamine*).

Kera'

Kera ikoon voldib kinni vaateakna (kui see sõlmel olemas on) (vt pilti *Kera*).

Kui kera on **punane**, võib selleks olla kolm põhjust:

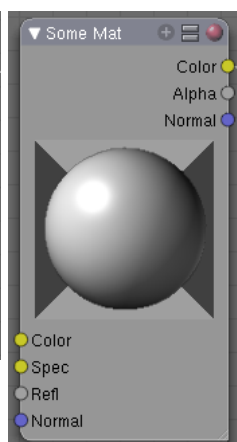
- See on [komposiitoris](#) oleva komposiitimisõlme Composite ainuke kasutatav väljund.
- See on ainuke [materjali](#) väljundisõlm Output (esimene, mis lisati).
- See on [materjali sisendisõlm](#), millele on määratud materjal (MA:).



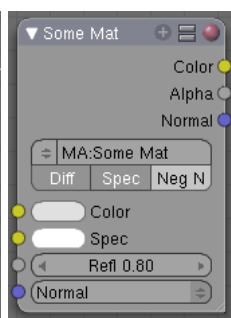
Kokkuvoltimise nool.



Plussmärk.



Menüü voltimine.

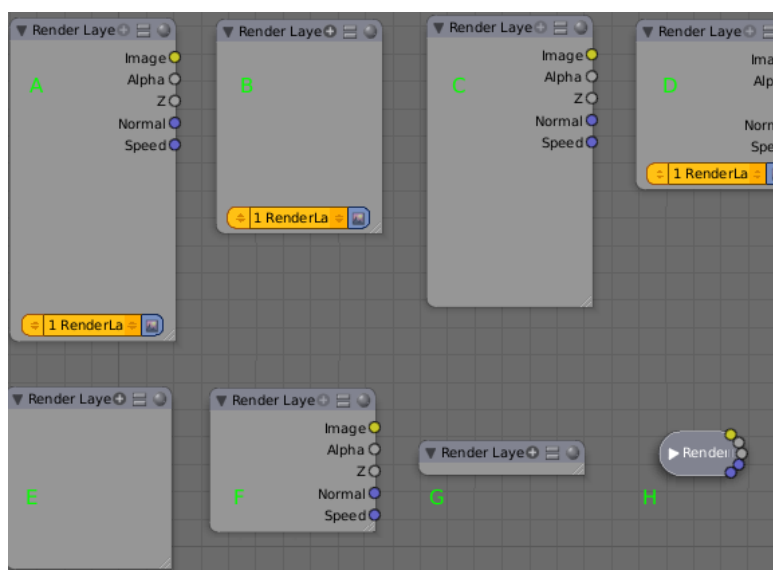


Kera.



Üheskoos.


Viimast kolme võib kasutada omavahel erinevates kombinatsioonides. Noolt, mis lükkab kokku terve sõlme, saab kasutada ainult koos plussmärgiga (vt pilti *Üheskoos*).



Sõlme ülemises ääres olevad suuruse muutmise kontrollid

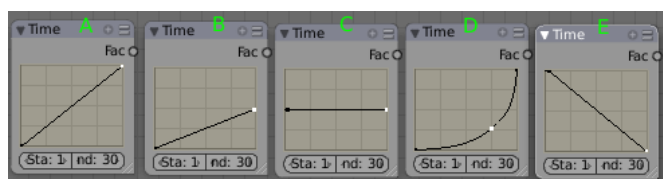
- A) Tavaline, B) Vajutatud on + märki, C) Vajutatud on = märki, D) Vajutatud on kera, E) Vajutatud on + ja = märke, F) Vajutatud on = märki ja kera G) Vajutatud on kõiki kolme H) Vajutatud on noolt.

Sõlme suuruse muutmine


Täpsemalt saab sõlme suurust mingil määral muuta, vajutades selle alumisel paremal nurgal (kus on väikesed kaldjooned) klahvi LMB  ja lohistades hiirt.

Sõlmekõverad

Mõnedel sõlmedel on kõveratega piirkonnad, mis tõlgivad sisendi väärtuse väljundi väärtuseks. Seda kõverat saab muuta, vajutades mõnele kontrollpunktile ja seda liigutades või kontrollpunkte lisades. Mõned näited on toodud allpool:



Sõlmekõvera muutmine.

Iga kõver on alguses sirge joon tõusuga 1. Sellisel kõveral on mõlemas otsas kaks väikest musta värvi juhtpunkti. Kui vajutada juhtpunkti peal LMB , siis see tähistatakse ja muutub valgeks.

Kõvera muutmine määrab, kuidas väljundit genereeritakse. Sisend, X, liigub tavaliselt lineaarselt (korrapärase vahemaadega) mööda **alumist** telge. Liigu üles kõverani ning sealt edasi **paremale**: see määrab vastava X-i väljundi Y. Seega teise näite puhul, kus X liigub mööda alumist serva punktist 0 punktini 1.0, varieerub Y-i väärtus 0.0 ja 0.5 vahel. Kolmandas, kus X liigub mööda alumist serva punktist 0.0 punktini 1.0, jääb Y-i väärtuseks pidevalt 0.5. Seega on ülemisel pildil olevatel kõveratel mõju ajale järgmine: **A** ei

mõjuta, **B** aeglustab, **C** peatab, **D** kiirendab ja **E** pöörab aja ringi.

"Kõverate" vidin on Blenderi kasutajaliidesesse sisse ehitatud ning seda saab igal pool kasutada tingimusel, et vidinale antakse vajalikud kõvera andmed. Praegu kasutatakse seda sõlmeredaktoris ja UV aknas.


See vidin kaardistab sisendi väärtuse horisontaalselt ning tekitab kõvera kõrgusest sõltuva uue väärtuse.

Märkus: see, et igal pildil on üks kõvera punkt "valge", *ei oma* tähtsust: see tähistab lihtsalt seda punkti, mille seda õpetust ettevalmistanud autor viimati valis. Antud juhul on oluline kõvera kuju, mitte selle määramiseks kasutatud kontrollpunktide asukoht (või nende värv).




RGB-kõverad

Ühes kõveravidinas saab muuta mitut erinevat kõverat. Selle tavaliseks näiteks on RGB-kõver, mille esimeseks kõveraks on "kombineeritud" tulemus või "värv" ("C") ning milles on olemas ka eraldi kõverad iga RGB komponendi (punane-roheline-sinine) muutmiseks. Kõik neli kõverat on korraga aktiivsed, kuid esimesena kasutatakse arvutamiseks "C" kõverat.

Kõverapunktide valimine


- LMB  valib alati ühe punkti ja tühistab ülejäänute valiku.
- Hoia vajutamise ajal all nuppu ⇧ Shift, et valitud punktide arvu suurendada või vähendada.

Kõverate muutmine

- LMB  vajutamine ja punkti lohistamine liigutab punkte.
- Kõvera peal LMB  vajutamine lisab uue punkti.
- Ühe punkti täpselt teise peale lohistamine liidab nad üheks punktiks.
- Lohistamise ajal ⇧ Shift all hoidmisel nakkuvad punktid ruudustikuga.
- Ctrl LMB  lisab punkti.
- Valitud punktide kustutamiseks vajuta ikooni X.

Vaate muutmine

Vaikimisi vaade keskendub 0.0–1.0 alale. Kui vaatekauguse piiraja on sisse lülitatud (nagu see vaikimisi on), ei saa vaadet lohistada ega seda suurendada/vähendada. Vaatekauguse piiraja välja lülitamiseks vajuta ikooni, mis meenutab märki #.

- LMB  vajutamine kõverast väljaspool ja hiire lohistamine liigutab vaadet
- Suurendamiseks ja vähendamiseks kasuta + ja - ikoone.

Eritööriistad

Mutrivõtme ikooni vajutamisel ilmub menüü, kust saad vaate algseks tagasi muuta, määrata punktide interpolatsiooni ning kõverat lähtestada.

Sõlmegrupid

Kui su kaart kasvab, võib olla sõlmede organiseerimisel abi mõnede sõlmede grupeerimisest. Väikese nõuandena soovitame paigutada sõlmede 'ootmisliin' vasakult paremale, ülevalt alla. Allpool seletame ka, kuidas sõlmi grupeerida alamprogrammideks.

Sõlmede grupeerimine võib muuta sõlmede paigutust sõlmeredaktoris lihtsamaks ning muuta töö materjali või komposiitmise 'nuudliga' (sõlmede võrgustikuga) kergemaks. Sõlmede grupeerimine loob samuti sõlmegruppe (.blend-faili sees olevad NodeGroupid) või sõlmepuid (NodeTreed, mida kasutatakse lisamisel).

Kontseptuaalselt laseb "grupeerimine" määrata sõlmede *komplekti*, mida võid kasutada samamoodi, nagu see oleks "lihtsalt üks sõlm." Seejärel saad seda gruppi samas või mõnes teises .blend-failis uuesti kasutada.

Näide: kui oled loonud materjali, mida sooviksid kasutada teises .blend-failis, *võiksid* sa lihtsalt selle materjali ühest .blend-failist teise tõsta. Aga kui sa sooviksid luua uue materjali ning kasutada selle juures juba mõne valmis sõlmede võrgustiku haru? Sa võiksid selle haru taasluua. Või saaksid selle materjali uude .blend-faili lisada ning seejärel kleepida selle materjali sisse soovitud haru. Mõlemad võimalused töötavad, kuid ei ole erinevate .blend-failidega töötamisel just eriti efektiivsed. Mis siis, kui looksid "sügavusteravuse" komposiitmissõlmede võrgustiku ja sooviksid seda teises .blend-failis kasutada? Või kui sooviksid ühte ja sama tegevusteseeriat tosin korda rakendada? Ka sellisel juhul *võiksid* sa selle võrgustiku uuesti luua, kuid see poleks eriti efektiivne tegevus. Paremad meetod nii materjali sõlmeharude kui komposiitmissõlmede võrgustike taaskasutamiseks oleks sõlmegruppide loomine.

Kui grupp määratakse, muutub see kindlaks objektiks; taaskasutatavaks tarkvarakomponendiks. Kui soovid, võid eirata implementatsiooni detaile ning kasutada seda (täpselt nii palju kordi kui soovid), milleks tarvis. Gruppe saab teistele failidele kasutatavaks muuta läbi [Blenderi teegi ning tavalise lisamise meetodi](#).

"PimedusseSulandumise" komposiitimisgrupp koosneb aja sõlmest (*Time*), mis annaks infot faktori (*Factor*) ja RGB-kõvera (RGB Curve) sõlmedele. RGB-kõvera sõlme kõver oleks nullis lame joon. Täpsemalt: iga sissetuleva X-i tulemuseks oleks nullväljund (must). Faktor algaks nulliga (sisendpilt ei muudeta) ja tõuseks üheni (täielik efekt ehk must). Sisendklipp sulanduks mustaks vastavalt aja sõlmes määratud kaadrite arvule.

Grupeerimine

Sõlmede grupi loomiseks vali sõlmeredaktoris sõlmed, mida soovid lisada, ja vajuta CtrlG või Space → node → make group.

Sõlmegrupi pealkirja riba on roheline. Kõik valitud sõlmed tehakse väikesteks ja pannakse grupi sisse.

Sõlmegruppidele antakse vaikimisi nimed *NodeGroup*, *NodeGroup.001* jne. Sõlmegrupis on nime väli, mille sisse vajutamisel saad grupi nime muuta. Nimeta sõlmegrupid ümber millekski tähenduslikuks. Sõlmegruppide lisamisel ühest .blend-failist teise ei erista Blender materjalisõlmede ja komposiitmissõlmede gruppe, seega soovitan võtta kasutusele mingi kindel nimetamissüsteem, mille alusel saaksid neid grupi tüüpe lihtsalt eristada. Näiteks materjalisõlmede harusid võid nimetada *Mat_XXX* ja komposiitmissõlmede võrke *Kom_XXX*.

Mida EI MAKSA gruppidesse lisada.

Pea meeles, et peamine idee seisneb selles, et iga grupp peaks põhimõtteliselt olema lihtsasti taaskasutatav, iseseisev komponent.

Materjalisõlmede grupid ei tohiks sisaldada materjali ja väljundi sõlmi.

Kui lisad gruppi materjali sõlme, siis ilmub materjalisõlm *kaks korda*: üks kord grupi sees ja teine kord grupist väljas uue materjali sõlmevõrgus.

Kui sa lisad grupi *sisse* väljundisõlme, siis ei tule grupist *välja* ühtegi väljundipesa!

Komposiitmissõlmede grupid ei saa sisaldada renderduskihi sõlme *Render Layer* (Blender ei lase sul seda sinna lisada) ning ei tohiks sisaldada väljundisõlme *Composite*. Ka siin tähendab lisatud väljundi sõlm (*Viewer*, *Split Viewer* jne) seda, et grupil ei ole enam väljundipesa.

Sõlmegruppide muutmine

Kui sõlmegrupp on valitud, muudab Tabi vajutamine sõlme aknaavaks ning näitatakse selles sisalduvaid üksikuid sõlmi. Neid saab ringi liigutada, üksikuid kontrollereid muuta, nende vahelisi siseseid lõimi ümber tõsta jne. Täpselt nii, nagu saaksid teha siis, kui nad oleksid sinu tavapärasel redaktoriaknas. Sa ei saa neid ühendada lõimega mõne grupist välja jääva sõlme külge, selleks pead sa kasutama grupisõlme külgedel olevaid väliseid pesi. Sõlmede gruppi lisamiseks või sealt eemaldamiseks pead grupi lõhkuma.

Sõlmegrupi lõhkumine

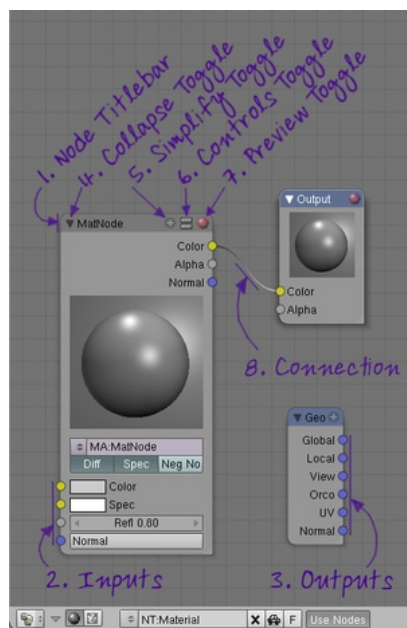
Kiirvalik AltG hävitab grupi ja asetab üksikud sõlmed tagasi redaktorisse. Kõik sisemised ühendused jäävad alles ning nüüd saad sa ka sisemisi sõlmi teiste redaktoris olevate sõlmedega ühendada.

Sõlmegruppide lisamine

Kui oled mõne sõlmepuu (*NodeTree*) oma .blend-faili lisanud, saad seda sõlmeredaktoris kasutada, vajutades Space → Add → Groups ning valides seejärel soovitud grupi. Grupi "juhtpaneeliks" on iga grupi sees oleva sõlme enda kontrollid. Neid saad grupiga töötamisel muuta nagu iga teise sõlme omasidki.

Sõlme juhtelemendid

See lehekülg selgitab sõlme juhtelemente



Sõlme peamised juhtelemendid

Nimeriba

Selles on sõlme nimi koos erinevate sõlme kokku voltimise nuppudega.

Sisendi pesad

Sõlme vasakul küljel on sisendi pesad:

- *sinised pesad* võtavad vastu vektoreid
- *kollased pesad* võtavad vastu värve
- *hallid pesad* võtavad vastu üksikväärtusi (nagu alfa)

Väljundi pesad

Sõlme paremal küljel on väljundi pesad:

- *sinised pesad* toodavad vektoreid
- *kollased pesad* toodavad värve
- *hallid pesad* toodavad üksikväärtusi (nagu alfa)

Pildi eelvaade / kõver

Sõlme sees on ala, mis näitab sõlmest väljuva pildi eelvaadet või kõveraid, mille abil saab määrata sõlme käitumist (näiteks RGB-sõlmes).

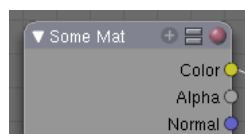
Nupud ja menüüd

Pildi eelvaate all on nupud ja menüüd, mis määravad sõlme käitumist.

Lõimed

Kõver joon kujutab endast väljundi pesa ja sisendi pesa vahelist ühendust. Pesade tüübid peavad olema vastavuses. Aktiivse sõlmega seotud ühendused on parema nägemise nimel esile tõstetud.

Kokku voltimise nupud



Sõlme päis.

Sõlme päises on kuni neli nähtavat juhtelementi (vt pilt *Sõlme päis*). Nendele vajutamine mõjutab seda, palju infot sõlm kuvab.

Sõlme nupp (▼►)

Vasakul olev nool voldib sõlme täielikult kokku või lahti.

Pesa nupp (+)

Paremal olev plussmärgiga nupp näitab/peidab kasutamata sisendi/väljundi pesasid.

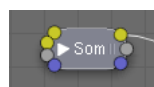
Menüü nupp (≡)

Kahe joonega nupp keskel näitab/peidab kõiki sõlmes olevaid juhtelemente.

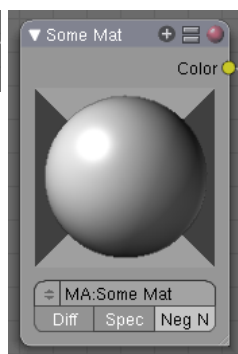
Eelvaate pildi nupp ()

Nimeriba kõige paremas servas olev keraga nupp peidab/näitab eelvaate pilti. The sphere button on the far right of the titlebar hides/unhides preview image. Kui kera on **punane** võib sel olla kolm põhjust:

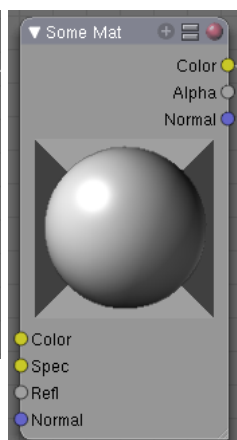
- see on ainus sõlmedaktori töötav väljundi sõlm.
- see on [materjali sisendi sõlm \(Material\)](#), millele on määratud materjal (MA:).



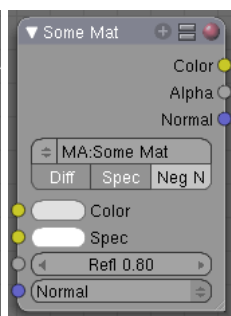
Kokku voltimise nool.



Plussmärk.



Menüü nupp.

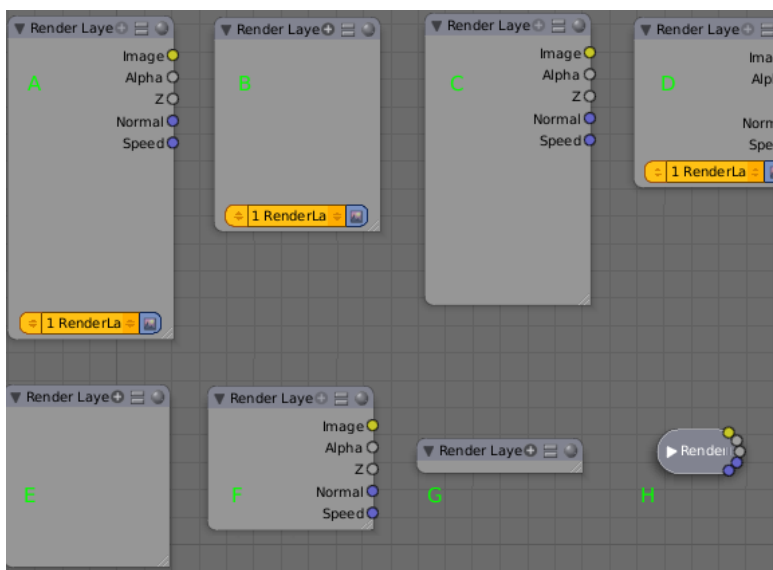


Kera.



Koos.


Viimast kolme saab omavahel koos kasutada. Noolt, mis voldib kokku terve sõlme, saab kasutada ainult koos plussmärgiga (vt pilt *Koos*).



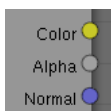
Sõlme ülemised suuruse nupud

- A) Tavaline, B) + märki on vajutatud, C) = märki on vajutatud, D) Kera on vajutatud, E) + ja = märke on vajutatud, F) = ja kera on vajutatud, G) Kõiki kolme on vajutatud H) Noolt on vajutatud.

Sõlme suuruse muutmine

Täpsemalt saab sõlme suurust mingil määral muuta, vajutades selle alumisel paremal nurgal (kus on väikesed kaldjooned) klahvi LMB  ja lohistades hiirt.

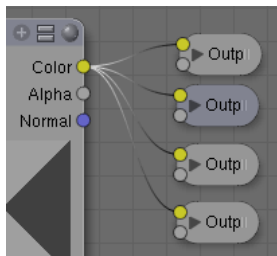
Pesad



Sõlme pesad.

Igal sõlmede aknas asuval sõlmel on "pesad" (sageli nimetatakse neid ka "ühendusteks" või sisendiks-väljundiks), mis kujutavad endast väikeseid värvilisi ringe, millesse ühendatakse sisend- ja väljundandmed (vt pilt *Sõlme pesad*).

Sõlme vasakul küljel olevad pesad tähistavad *sisendeid*, *paremale jäävad pesad väljundeid*.



Sõlmede ühendamine.

Elu lihtsustamiseks on pesad värviga tähistatud vastavalt sellele, mis tüüpi informatsiooni nad eelistavad saata või vastu võtta. Värv on kolm:

Kollased pesad

Tähistab seda, et pesa võetakse vastu või sellest saadetakse sõlme **värvi** informatsiooni.

Hallid pesad

Tähistab väärtustega (**numbrilist**) informatsiooni. See võib olla kas üksik numbriline väärtus või "väärtuste kaart". (Väärtuste kaarti võid ette kujutada halltoonides kaardina, kus erinev heleduse/tumeduse hulk väljendab iga eri punkti väärtust.) Kui "väärtuste kaardi" pesas kasutada üksikut väärtust, kasutavad kõik kaardi punktid sama väärtust. Tavaline kasutamisiis: alfakaardid ja sõlmede numbrilise väärtusega väljad.

Sinised/lillad pesad

Tähistab **vektor-/koordinaat-/normaalide** informatsiooni.

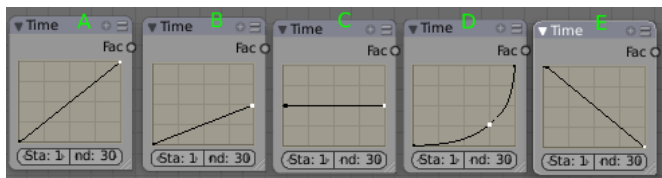
Kui sa ei kasuta *konverterit*, millest räägime hiljem, tuleb sõlmede vahel kollased pesad ühendada kollastega, hallid hallide ja sinised sinistega.

Värvi kõrval näed sõlmes vastava pesa nime. Kuigi see ei ole alati nii, võib nime pidada vihjeks selle kohta, mis laadi infot pesa eeldab. Kuid see ei tähenda, et info *peab* seda tüüpi olema. Ma võin näiteks ühendada alfata sisaldava pesa nimega (Alpha) halli värvi pesa mõne materjalisõlme halli peegelduvuse (Reflection) pesa ning saavutada ikkagi tulemust. Kõige olulisem on selle puhul, et tegemist on "hall-hall" ühendusega.

On erandeid, mille puhul saad ilma konverteriteta segada kollast (*nt* värviline pilt) ja halli (*nt* halltoonid). Tavaliselt lisab Blender ise konverteri, kui seda vaja on, seega eksperimenteerige vabalt. Saad kasutada vaataja väljundisõlmi "Viewer", mida seletame hiljem, nägemaks, kas ja kuidas su eksperiment töötab.

Kõverad

Mõnedel sõlmedel on kõveratega piirkonnad, mis tõlgivad sisendi väärtuse väljundi väärtuseks. Seda kõverat saab muuta, vajutades mõnele kontrollpunktile ja seda liigutades või kontrollpunkte lisades. Mõned näited on toodud allpool:



Sõlme kõvera muutmine.

Iga kõver on algses sirge joon tõusuga 1. Sellisel kõveral on mõlemas otsas kaks väikest musta värvi juhtpunkti. Kui vajutada juhtpunkti peal LMB , siis see tähistatakse ja muutub valgeks.

Kõvera muutmine määrab, kuidas väljundit genereeritakse. Sisend, X, liigub tavaliselt lineaarselt (korrapäraste vahemaadega) mööda **alumist** telge. Liigu üles kõverani ning sealt edasi **paremale**: see määrab vastava X-i väljundi Y. Seega teise näite puhul, kus X liigub mööda alumist serva punktist 0 punktini 1.0, varieerub Y-i väärtus 0.0 ja 0.5 vahel. Kolmandas, kus X liigub mööda alumist serva punktist 0.0 punktini 1.0, jääb Y-i väärtuseks pidevalt 0.5. Seega on ülemisel pildil olevatel kõveratel mõju ajale järgmine: **A** ei mõjuta, **B** aeglustab, **C** peatab, **D** kiirendab ja **E** pöörab aja ringi.

"Kõverate" vidin on Blenderi kasutajaliidesesse sisse ehitatud ning seda saab igal pool kasutada tingimusel, et vidinale antakse vajalikud kõvera andmed. Praegu kasutatakse seda sõlmede redaktoris ja UV aknas.

See vidin kaardistab sisendi väärtuse horisontaalselt ning tekitab kõvera kõrgusest sõltuva uue väärtuse.

Märkus: see, et igal pildil on üks kõvera punkt "valge", *ei oma* tähtsust: see tähistab lihtsalt seda punkti, mille seda õpetust ettevalmistanud autor viimati valis. Antud juhul on oluline *kõvera* kuju, mitte selle määramiseks kasutatud kontrollpunktide asukoht (või nende värv).




RGB-kõverad

Ühes kõveravidinas saab muuta mitut erinevat kõverat. Selle tavaliseks näiteks on RGB-kõver, millel esimeseks kõveraks on "kombineeritud" tulemus või "värvi" ("C") ning milles on olemas ka eraldi kõverad iga RGB komponendi (punane-roheline-sinine) muutmiseks. Kõik neli kõverat on korraga aktiivsed, kuid esimesena kasutatakse arvutamiseks "C" kõverat.

Kõverapunktide valimine


- LMB valib alati ühe punkti ja tähistab ülejäänute valiku.
- Hoia vajutamise ajal all nuppu Shift, et valitud punktide arvu suurendada või vähendada.

Kõverate muutmine

- LMB  vajutamine ja punkti lohistamine liigutab punkte.
- Kõvera peal LMB  vajutamine lisab uue punkti.
- Ühe punkti täpselt teise peale lohistamine liidab nad üheks punktiks.
- Lohistamise ajal ⇧ Shift all hoidmisel nakkuvad punktid ruudustikuga.
- Ctrl LMB  lisab punkti.
- Valitud punktide kustutamiseks vajuta ikooni X.

Vaate muutmine

Vaikimisi vaade keskendub 0.0–1.0 alale. Kui vaatekauguse piiraja on sisse lülitatud (nagu see vaikimisi on), ei saa vaadet lohistada ega seda suurendada/vähendada. Vaatekauguse piiraja välja lülitamiseks vajuta ikooni, mis meenutab märki #.

- LMB  vajutamine kõverast väljaspool ja hiire lohistamine liigutab vaadet
- Suurendamiseks ja vähendamiseks kasuta + ja - ikoone.

Eritööriistad

Mutrivõtme ikooni vajutamisel ilmub menüü, kust saad vaate algseks tagasi muuta, määrata punktide interpolatsiooni ning kõverat lähtestada.

Sõlmede kasutamine

Sõlmede lisamine


Sõlmeredaktorisse lisatakse sõlmi kahel moel:

- Kasutades sõlmeredaktori tööriistaribal olevat lisamise menüüd Add ja valides sealt soovitud sõlmetüübi või
- Hoides kursorit sõlmeredaktori aknas, vajutades kiirvalikut ⇧ ShiftA ning valides hüpikmenüüst Add mõne sõlme.

Sõlmede paigutamine

Üldiselt proovi paigutada sõlmi aknas nii, et pildi arvutamine toimuks vasakult paremale, ülalt alla. Sõlme liigutamiseks vajuta mõnele selle neutraalsele alale ja lohista siis ringi. Sõlme peale võib vajutada peaaegu ükskõik kuhu ning seejärel seda ringi lohistada; ühendused proovivad ennast võimalikult hästi Bezier' kõveratena ümber kohendada.



Sõlmede ühendamine

Vajuta soovitud pesale klahviga LMB  ja lohista hiirt: näed, kuidas pesast tuleb välja haru. Seda kutsutakse "lõimeks".

Jätka lohistamist ja ühenda see lõime mõne teise sõlme sisendi pesaga, lastes seejärel klahvi LMB  lahti.

Sellisel juhul saadetakse iga väljundpesa edasi mööda sõlmi. Kuid igasse sisendpesasse saab ühenduda ainult üks lõim.

Sõlmede lahtiühendamine

Pesadevahelise ühenduse katkestamiseks vajuta lõigatava lõime lähedale tühjale alale klahviga LMB  ja lohista hiirt - hiire kursori kõrvale ilmub pisike noatera ikoon. Liiguta hiir üle lõigatava lõime ning lase siis klahv LMB  lahti.

Sõlmede duplitseerimine

Vajuta soovitud sõlmele, seejärel kasuta kiirvalikut ⇧ ShiftD ning liiguta hiirt, et hiirekursori alla ilmuks valitud sõlme koopia.

Ahhaa!

Kui sa mõnd sõlme duplitseerid, siis asetatakse uus sõlm *täpselt* vana peale. Kui sõlme sinna jätadki (ja seda võib julgelt teha), siis sa lihtsalt ei näe, et seal on tegelikult *kaks* sõlme! Jah, sõlme vari võib olla natuke tumedam kui enne, kuid enam ei ole see visuaalselt selge, mis on millega ühendatud. Kui tekib kahtlusi, haara sõlmest kinni ja liiguta seda natuke, et näha, kas selle all miski ennast varjab.

Sõlmegrupid

Sõlmede grupeerimine võib muuta sõlmede paigutust sõlmeredaktoris lihtsamaks ning muuta töö materjali või komposiitmise 'nuudliga' (sõlmede võrgustikuga) kergemaks. Sõlmede grupeerimine loob samuti sõlmegruppe (.blend-faili sees olevad NodeGroupid) või sõlmepuid (NodeTreed, mida kasutatakse lisamisel).

Kontseptuaalselt laseb "grupeerimine" määrata sõlmede *komplekti*, mida võid kasutada samamoodi, nagu see oleks "lihtsalt üks sõlm." Seejärel saad seda gruppi samas või mõnes teises .blend-failis uuesti kasutada.

Näide: kui oled loonud materjali, mida sooviksid kasutada teises .blend-failis, *võiksid* sa lihtsalt selle materjali ühest .blend-failist teise tõsta. Aga kui sa sooviksid luua uue materjali ning kasutada selle juures juba mõne valmis sõlmede võrgustiku haru? Sa võiksid selle haru taasluua. Või saaksid selle materjali uude .blend-faili lisada ning seejärel kleepida selle materjali sisse soovitud haru. Mõlemad võimalused töötavad, kuid ei ole erinevate .blend-failidega töötamisel just eriti efektiivsed. Mis siis, kui looksid "sügavusteravuse" komposiitmissõlmede võrgustiku ja sooviksid seda teises .blend-failis kasutada? Või kui sooviksid ühte ja sama tegevusteseeriat tosin korda rakendada? Ka sellisel juhul *võiksid* sa selle võrgustiku uuesti luua, kuid see poleks eriti efektiivne tegevus. Paremad meetod nii materjali sõlmeharude kui komposiitmissõlmede võrgustike taaskasutamiseks oleks sõlmegruppide loomine.

Kui grupp määratakse, muutub see kindlaks objektiks; taaskasutatavaks tarkvarakomponendiks. Kui soovid, võid eirata implementatsiooni detaile ning kasutada seda (täpselt nii palju kordi kui soovid), milleks tarvis. Gruppe saab teistele failidele kasutatavaks muuta läbi [Blenderi teegi ning tavalise lisamise meetodi](#).

Grupeerimine

Panel: Sõlmeredaktor

Menu: ⇧ ShiftA → Groups (grupid)

Sõlmede grupi loomiseks vali sõlmeredaktoris sõlmed, mida soovid lisada, ja vajuta CtrlG või Space → *node* → *make group*. Sõlmegrupi pealkirja riba on roheline. Kõik valitud sõlmed tehakse väikesteks ja pannakse grupi sisse. Sõlmegruppidele antakse vaikimisi nimed *NodeGroup*, *NodeGroup.001* jne. Sõlmegrupis on nime väli, mille sisse vajutamisel saad grupi nime muuta. Nimeta sõlmegrupid ümber millekski tähenduslikuks. Sõlmegruppide lisamisel ühest .blend-failist teise ei erista Blender materjalisõlmede ja komposiitmissõlmede gruppe, seega soovitan võtta kasutusele mingi kindel nimetamissüsteem, mille alusel saaksid neid grupi tüüpe lihtsalt eristada. Näiteks materjalisõlmede harusid võid nimetada *Mat_XXX* ja komposiitmissõlmede võrke *Kom_XXX*.



Mida EI MAKSA gruppidesse lisada (kõigi sõlmeredaktori tüüpide puhul)

Pea meeles, et peamine idee seisneb selles, et iga grupp peaks põhimõtteliselt olema lihtsasti taaskasutatav, iseseisev komponent. Sõlmede grupid **ei tohiks sisaldada**:

Allikasõlmi

Kui lisad grupi allika sõlme, siis ilmub allika sõlm *kaks korda*: üks kord grupi sees ja teine kord grupist väljas uues sõlmevõrgus.

Allika sõlmed on näiteks: *Materjali sõlm* (materjalide sõlmeredaktoris) ja *Rendersukihtide sõlm* (komposiitimisredaktoris).

Väljundi sõlme

Kui sa lisad grupi *sisse* väljundisõlme, siis ei tule grupist *välja* ühtegi väljundipesa!

Väljundi sõlmed on näiteks: väljundi sõlm *Output* (materjalide sõlmeredaktoris) ja vaataja sõlm *Viewer* (komposiitimisredaktoris).

Sõlmegruppide muutmine

Kui sõlmegrupp on valitud, muudab ⇆ Tab vajutamine sõlme aknaavaks ning näitatakse selles sisalduvaid üksikuid sõlmi. Neid saab ringi liigutada, üksikuid kontrollereid muuta, nende vahelisi siseseid lõimi ümber tõsta jne. Täpselt nii, nagu saaksid teha siis, kui nad oleksid sinu tavapärase redaktoriaknas. Sa ei saa neid ühendada lõimega mõne grupist välja jääva sõlme külge, selleks pead sa kasutama grupisõlme külgedel olevaid väliseid pesi. Sõlmede gruppi lisamiseks või sealt eemaldamiseks pead grupi lõhkuma.

Sõlmegrupi lõhkumine

Kiirvalik AltG hävitab grupi ja asetab üksikud sõlmed tagasi redaktoris. Kõik sisemised ühendused jäävad alles ning nüüd saad sa ka sisemisi sõlmi teiste redaktoris olevate sõlmedega ühendada.

Sõlmegruppide lisamine

Kui oled mõne sõlmepuu (*NodeTree*) oma .blend-faili lisanud, saad seda sõlmeredaktoris kasutada, vajutades Space → *Add* → *Groups* ning valides seejärel soovitud grupi. Grupi "juhtpaneeliks" on iga grupi sees oleva sõlme enda kontrollid. Neid saad grupiga töötamisel muuta nagu iga teise sõlme omasidki.

Komposiitimisõlmede tüübid

See osa manualist on organiseeritud sõlmetüüpide kaupa. Iga sõlmetüübi sõlmed sooritavad mingis mõttes samaseid operatsioone:

- [Input](#) - sisendsõlmed, võimaldavad sõlmekaarti tuua pilte, tekstuure, värve või väärtuseid.
- [Output](#) - väljundsõlmed, näitavad vahe- või lõpptulemusi ning võimaldavad tulemust salvestada.
- [Color](#) - värisõlmed, manipuleerivad pildi värvidega.
- [Vector](#) - vektorsõlmed, võimaldavad manipuleerimist vektorinfo kihtidega (näiteks pinnanormaalid).
- [Filters](#) - filtrisõlmed, töötlevad mitut pildi pikslit korraga, näiteks erinevad udustamised.
- [Convertors](#) - konvertersõlmed, võimaldavad pildikanalite eraldamist ja kombineerimist.
- [Mattes](#) - maskimissõlmed, meid huvitava objekti pildist eraldamiseks kas värviliselt taustalt või mõnel muul viisil.
- [Distortion](#) - moonutussõlmed, võimaldavad pilti nihutada, keerata-pöörata ning lisada näiteks läätsemoonutust.
- [Groups](#) - kasutaja poolt määratud sõlmegrupid.

Komposiitimise sisendisõlmed (*Input*)

Sisendisõlmed genereerivad erinevatest allikatest sisendpilte.

Sisendiks võib olla näiteks:

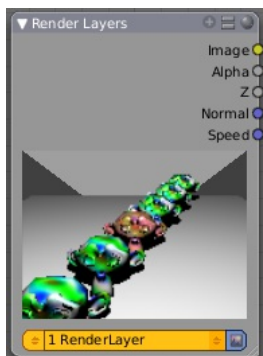
- aktiivsest stseenist renderdatud pilt,
- JPG, PNG vms formaadis pilt,
- videoklipp (näiteks animatsioon videofaili või pildijada kujul),
- tekstuur,
- genereeritud ühevärviline pilt või
- aja alusel genereeritud väärtus.

Need sõlmed loovad teistele sõlmedele sisendid. Seega pole neil endil sisendeid, on vaid väljundid.

Renderduskihtide sõlm (*Render Layers*)

Panel: [Sõlmeredaktor](#) → [Node Composition \(komposiitimine\)](#)

Menu: ⇧ ShiftA → [Input \(sisend\)](#) → *Render Layers* (renderduskihid)



Renderduskihtide sõlm
Render Layers

Selle sõlme abil saad oma stseenist renderdatud pildi sõlmekaardile.

Sõlm laeb aktiivsest kaamerast renderdatud pildi sisendina kasutamiseks. Vali sõlme seadetest (allservas) stseen ja aktiivne renderduskiht, mille pilti sa soovid kasutada. Blender kasutab valitud stseeni aktiivset kaamerat ja valitud [renderduskihi](#) (*RenderLayer*) objekte.

Väljund Image (pilt) sisaldab renderdatud pildi RGB-kanaleid

- Alpha on läbipaistvuse kanal

Sõltuvalt renderduskihi seadetest võivad kasutatavad olla ka teised väljundpesad. Vaikimisi on sisse lülitatud Z-kanal ehk sügavusinfo:

- Z-kanal (ehk info selle kohta, kui kaugel on iga konkreetne piksel kaamerast)

Pildil olevas näites on sisse lülitatud ka kaks lisakanalit:

- Normal pinnanormaali vektorid,
- Speed (kiirus) piksli nihkevektorid (mida saab kasutada liikumishägu loomiseks).

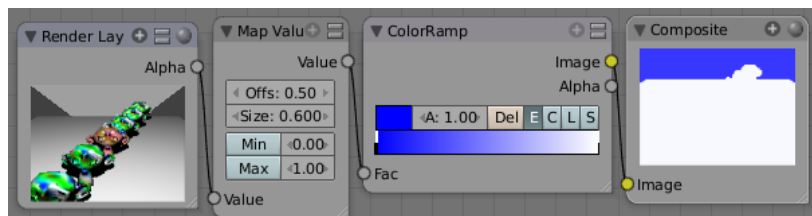
Kasuta renderdamise nuppu (väike fotoaparaadi pildiga ikoon renderduskihi nime kõrval), et stseeni uuesti renderdada ja pilti värskendada.

Tuleta meelde, et üks .blend-fail võib sisaldada mitmeid erinevaid stseene. *Renderlayer* sõlm võimaldab sul määrata, millist stseeni sa sisendiks tahada. Kui sa määrad sisendiks mõne teise stseeni, milles on samuti komposiitimine sisse lülitatud, siis sisendiks laetakse siiski pilt, mis on komposiitimata, otse kaamerast tulnud. Kui sa soovid sisendina laadida juba komposiititud pilti, siis pead sa selle salvestama pildifailiks, soovitatavalt OpenEXR-formaati, kuna see salvestab ka kõik lisakanalid.

Alpha-väljundi kasutamine

Läbipaistvuse (Alpha) väljundit läheb sul vaja, kui sa soovid pilte läbipaistvuse alusel üksteise peale kombineerida.

Blenderi stseenis asuvad objektid virtuaalses ruumis, millel pole vaikimisi tausta. Mõned objektid võivad paikneda teiste ees või taga, kuid siiski puudub stseenil vaikimisi kogu tagaplaani kattev taust. Stseeni seadetest võib taustaks lisada näilise horisondi, kuid see on vaid illusioon, mitte reaalne objekt. Mõned objektid on poolläbipaistvad. Läbipaistvuse info säilitatakse spetsiaalses piksliväärtuste kanalis nimega Alpha. Poolläbipaistev objekt võimaldab valgusel ja tagaplaani pildil läbi selle paista, muutes objekti taga oleva nähtavaks (nagu näiteks klaasist aken). Stseeni renderdades genereerib Blender lisaks lõpp-pildile ka pildi, mille piksliväärtused kannavad objektide läbipaistvuse infot. Seda läbipaistvuse kanalit saad sa silmaga näha, ühendades ta näiteks komposiitpildi pildisissendisse (*Image*) või vahekuva sõlme (*Viewer*) sisendisse:



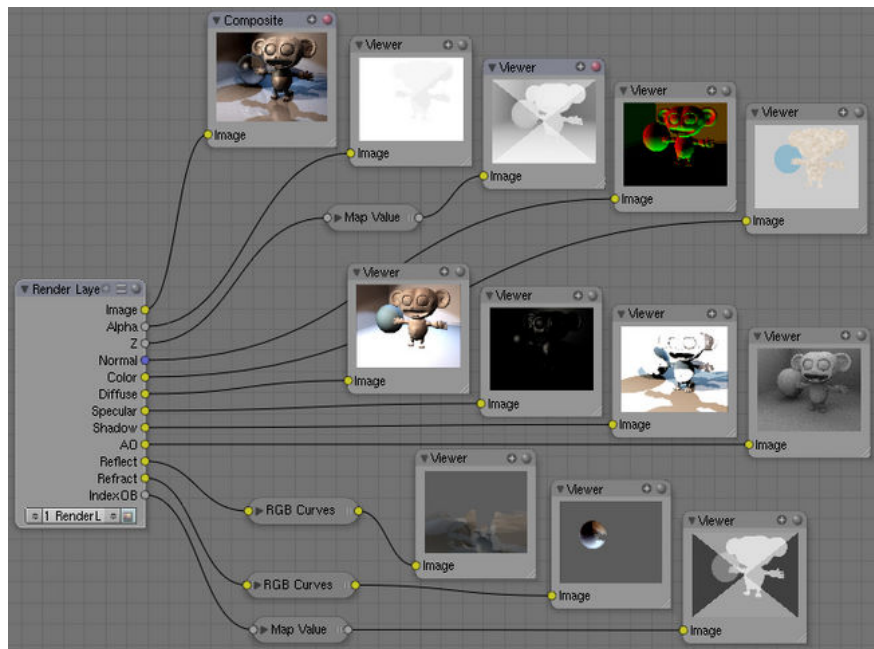
Alfakanalite kuvamine

Siin näites kasutame me alfakanali visualiseerimiseks tema toonimist. Alfakanali väljund on ühendatud *Map Value* sõlme. *Color Ramp* sõlme abil anname me igale pikslile vastavalt tema väärtusele uue värvi. *Color Ramp* sõlme väljund on omakorda ühendatud komposiitpildi sõlme (*Composite*), et näidata meile lõplikku tulemust. Pane tähele, et tulemusena kuvatakse meile täielikult läbipaistmatud alad valge värviga ning läbipaistvad (sealhulgas ka taust, kus objekte ei ole) sinise värviga.

Valikulised väljundid

Valikuliste väljundite lisamiseks tuleb vastavad renderduskäigud (*Passes*) renderdamise seadete alt sisse lülitada. Seda saab teha renderdamise kaardi paneelil *Layer* (kiht) olevate linnukesekastide abil. Näiteks varjude renderduskäigu (*Shadow*) kasutamiseks peab olema lisatud linnuke kasti *Shadow*(vari). Tutvu erinevate võimalike renderduskäikudega renderdamise kihi seadete alt. Pane tähele, et igale renderduskihile on võimalik määrata ka erinevad renderduskäigud.

Juuresoleval pildil on kujutatud lihtsa stseeni renderdus koos kõigi võimalike renderduskäikudega:



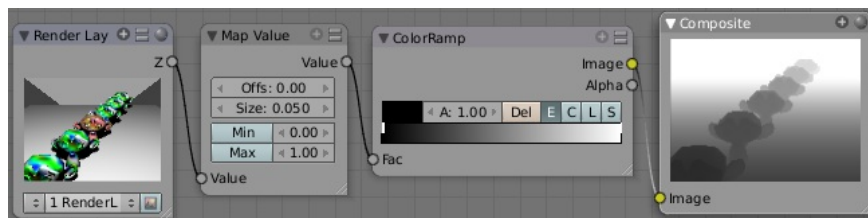
Kasutatavad renderduskäigud on:

- *Z*: kaugus kaamerast, Blenderi ühikutes
- *Normal* (*Nor*): normaali vektor - kuidas küljelt tulev valgus värve mõjutav
- *UV*: tekstuurikoordinaadid - kuidas UV-laotus pilti moonutab
- *Speed* (*Vec*): nihkevektor - millises suunas ja kui kiiresti objekt liigub
- *Color* (*Col*): sinu poolt nähtava pildi RGB-väärtused
- *Diffuse*: pindade hajuspeegelduste kiht - kuidas materjalist läbi hajuvad värvid pehmenevad
- *Specular*: läike kiht - kui palju objektidele valguse käes läiget lisatakse
- *Shadow*: varjude kiht - kuidas objektid teistele objektidele varje heidavad
- *AO*: kaudvalguse varjutuse kiht - kuidas maailma valgustus värve mõjutav
- *Reflect* (*Ref*): peegelduste kiht - kuidas peegeldavad objektid oma baasmaterjalist hoolimata värve peegeldavad
- *Refract*: murdunud valguse kiht - kuidas värvid läbipaistvaid objekte läbides murduvad
- *Radio* (*Radiosity*): objektide poolt kiiratud valguse kiht - kuidas objektid stseeni värve kiirgavad
- *IndexOB*: objektide ID koodide kiht - kaamera poolt nähtavate objektide indeks

Z-kanali kasutamine

Z-kanalit ehk sügavusinfot on võimalik kasutada mitmel erineval viisil, enamasti leiab ta rakendust sügavusudu loomisel.

Kujuta ette kaamerat, mille vaateväli on risti X-Y tasandiga. Läbi kaamera vaadates on Y-telg üles-alla ning X-telg vasakule-paremale suunaga. Kaamerast tasandi poole, tasandiga risti on Z-telg, mille nullpunkt asub kaamera juures. Kõigil stseeni objektidel, mis jäävad kaamera ette, on kaamera suhtes positiivne Z-väärtus. Mida kaugemal nad on, seda suurem see on. Lisaks kanale pildile ja selle läbipaistvuse kanalile genereeritakse renderdamisel alati vaikimisi ka Z-kanal. Erinevalt pildikanalist võivad Z-kanalis piksliväärtused olla ka suuremad kui 1.0. Seda kanalit on võimalik näha, kui selle piksliväärtused skaleerida vahemikku 0.0 kuni 1.0 (näiteks sõlme *Map Value* abil):



Z-kanali sisu kuvamine

Sellel pisikesel sõlmekaardil on *RenderLayer* Z-väljund ühendatud *Map Value* sõlme. *Map Value* sõlm skaleerib kanali piksliväärtused määratud vahemikku. *Color Ramp* sõlm määrab igale väärtusele vastava värvi. Lõpptulemuse vaatamiseks on *Color Ramp* sõlme väljund ühendatud komposiitpildi kuvajasse (*Composite viewer*). Pane tähele, et sõlme *Map Value* sõlme seaded on määratud nii, et kaamerale lähemal olevad objektid on tumedamad ning kaugemad heledamad. Parameeter *Size* ehk skaleerimistegur on 0.05, mis tähendab, et me näeme pildil sügavusvahemikku 0 kuni 20 ühikut ($20 \times 0.05 = 1$ ehk valge värv).

Kiirusvektori ühenduspesa (*Speed*) kasutamine

Kuigi meil võivad stseenis objektid olla animeeritud, ei sisalda renderdatud üksikkaader mitte mingisugust infot liikumise kohta. Samas on võimalik *Render Layers* sõlme abil genereerida lisakanal, mille piksliväärtused sisaldavad infot pildi iga piksli kiirusvektori kohta antud kaadris. Kiirusvektori väljundit saad kasutada [liikumishägu lisamiseks](#). [Rohkem infot leiad siit](#).

Pildi lugemise sõlm (*Image*)

Panel: [Sõlmeredaktor](#) → [Node Composition \(komposiitimine\)](#)

Menu: ⇧ ShiftA → [Input \(sisend\)](#) → *Image* (pilt)

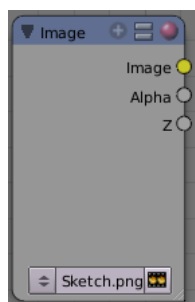
Pildi lugemise sõlm
Image

Image sõlm loeb pildifaili [Blenderi poolt toetatud formaadis](#). Peale pildi RGB väärtuste on võimalik lugeda ka Alpha (läbipaistvuse) kanali ja sügavusinfo (Z) kanali väärtusi, kui pilt neid sisaldab. Kui laetava pildi formaat võimaldab lisakanaleid (näiteks OpenEXR), siis laetakse kõik olemasolevad kanalid. Kasuta seda sõlme, et laadida:

- üksikut pilti või kaadrit (näiteks JPG pilti)
- animatsiooni kaadrijada või osa sellest (näiteks kaadrid 30-60)
- videofaili või selle osa (näiteks AVI video)
- pilti, mis on parasjagu avatud Blenderi UV/pildiredaktoris
- pilti, mis on sessiooni jooksul olnud avatud Blenderi UV/pildiredaktoris

Animatsiooni pildijada laadimise juhend: [Animatsioonid](#).

Pildifaili laadimiseks klõpsa väikesel pildiga nupul ning vali mõni stseenis sisalduv pilt või tekstuur. Pildi laadimiseks kõvakettalt klõpsa nupul Load (laadi) ja navigeeri failisirvijas soovitud failini. Laetav pilt võib olla mõni eelnevalt renderdatud kaader, joonistatud taust, pilt sinu kassist või mis iganes muu pildifail. Blenderit pildi sisu ei huvita.

Kui pilt on osa animatsioonist, vali pärast pildi laadimist rippmenüüst Source (allikas) valik *Image Sequence* (pildijada). Avanevad lisavalikud võimaldavad määrata, millist osa pildijadast sa laadida soovid. Kui tegemist on videofailiga, avanevad need valikud automaatselt.

Pildi kanalid

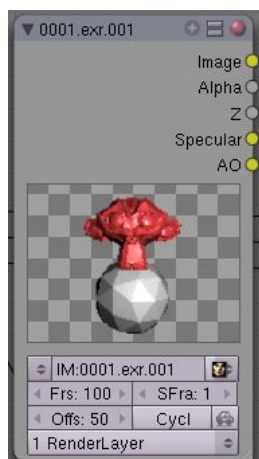
Pärast pildi laadimist ilmuvad kättesaadavad pildikanalid ühenduspesadena sõlme paremasse serva. Kindlasti peaks saadaval olema kanal *Image* ehk pilt ise. *Alpha* ehk läbipaistvuse kanali ja *Z* ehk sügavusinfo kanali olemasolu sõltub pildi formaadist. Kui pildifaili formaat ei toeta A ja/või Z kanaleid, määratakse neile vaikeväärtused (1.0 A jaoks, 0.0 Z kanali jaoks).

- Alfa ehk läbipaistvuse kanal
 - Kui sisendfailis sisaldub alfakanal, muutub väljundpesa Alpha aktiivseks.
 - Kui pildifailis pole alfakanalit, määratakse alfaks vaikeväärtus. (Vaikeväärtus on 1.0 - täiesti läbipaistmatu ning kui ühendada see Viewer sõlme sisendpessa, kuvatakse ta täiesti valgena).
- Z ehk sügavusinfo kanal
 - Kui sisendfailis sisaldub Z-kanal, muutub Z-väljundpesa aktiivseks.
 - Kui pildifailis pole Z-kanalit (näiteks JPG või PNG failid), määrab Blender selleks vaikeväärtuse (ehk 0.00, st kaamera tasandi). Z-kanali uurimiseks ühenda see *Map Value* ja *ColorRamp* sõlmede abil, nagu näidatud [renderduskihi \(Render Layer\) sisendsõlme lõigus](#).

Formaadid

Blender toetab paljusid pildiformaate. Praegusel hetkel toetab neist vaid OpenEXR-formaat RGB (värvinfo), A (alfa) ja Z (sügavus) puhvrite ühte faili salvestamist.

Renderduskäikude salvestamine ja laadimine



Blender võimaldab renderduskihte ja -käike salvestada nn MultiLayer-failiformaati, mis on OpenEXR-formaadi laiendus. Järgnevas näites avame me pildijadast kaadrite vahemiku 50 kuni 100. Pildifailidesse on salvestatud värvikanalid RGB, alfa kanal, Z-kanal, läike kiht (*Specular*) ning kaudvalguse varjutuse (*AO*) renderduskäik.

MultiLayer-tüüpi faili salvestamiseks tuleb renderdamise seadetest välja lülitada seade *Do Composite* (komposiidi), määrata formaadiks *MultiLayer* (mitmekihiline) ning lülitada sisse renderduskäigud, mida sa soovid faili salvestada. Faili uuesti avamiseks lülitatakse sisse *Compositing Nodes* ja *Do Composite* (komposiidi) ning lae pilt (või jada) sisendsõlme *Image* abil. Avatud failis peaksid olema kättesaadavad kõik sinna salvestatud renderduskäigud.

Pildi suurus

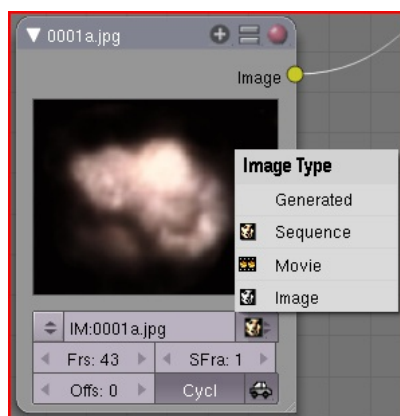
Suurus loeb - jälgi pildifailide lahutust ja värvisügavust, kui sa erinevaid faile kombineerid. Erineva suuruse ja värvisügavusega piltide komposiitimise tulemuseks võib olla pildi sakilisus, värvide "lamedus" ning väljavenitatud pildid.

Blenderi komposiitor suudab segada kõigis suurustes pilte, kuid need operatsioonid toimivad vaid seal, kus pildid kattuvad. Erineva suurusega sisendite puhul käituvad Blenderi sõlmed järgnevalt:

- Esimene (ülemine) sisendpilt määrab väljundi lahutuse.
- Vaikimisi on pildid tsentreeritud kaadri keskele, kui neid pole just nihkesõlme (*Translate*) abil liigutatud.

Seega suudavad kõik sõlmed opereerida erinevas suuruses piltidega ning tulemuse määravad sisendite formaadid. Vaid komposiitmissõlmel (*Composite*) on väljundi suurus määratud stseenikonteksti (*Scene*) alajaotuses *Dimensions* (mõõtmed). Kuvaja (*Viewer*) sõlm kuvab alati sisendi lahutusega pildi, kuid kui sisendit pole ühendatud (või on sisendiks konstantne väärtus), näitab ta 320×256 piksli suurust pilti.

Animatsioonid



Pildijada või videofaili avamiseks klõpsa nupul *Open* või väikesel kausta pildiga nupul ning vali kas jada esimene kaader või videofail. Pärast faili valimist saad sõlme rippmenüüst *Source* valida, mis tüüpi failiga on tegemist:

1. *Generated* - see tüüp on mõeldud Blenderi tekstuuride lisamiseks komposiitorisse. Selleks pead tekstuurid kõigepealt tekstuuride paneeli kaudu valmis meisterdama ning siis avama, klõpsates väikesele pildi ikooniga nupule.
2. *Sequence* - pildijada, animatsioon, kus iga kaader on eraldi pildifail.
3. *Movie* - videofail, .avi või .mov konteiner
4. *Image* - üksik pildifail

Pildi või video nimetamisel piiranguid pole, kuid pildijada laadimiseks peab faili nimes olema määratud tema järjekorranumber. Numbrikoht peab olema kõigil failidel sama pikk, seepärast tuleb puuduvad positsioonid täita nullidega. Näiteks kui meil on animatsiooni viimane kaader järjekorranumbriga 1000 (neli kohta), siis peab esimene kaader olema mitte *test_1.jpg*, vaid *test_0001.jpg*. Soovitav on panna järjekorranumber nimes kõige viimaseks ning eraldada ta ülejäänud nimest alakriipsuga.

Pildijada või video avamisel kuvab sõlm lisavõimalused sobiva lõigu lugemiseks. Kasuta neid seadeid, et määrata alguskaader, video pikkus ning kaadrite nihe. Sa võid laadida nii kogu video kui ka näiteks vaid sobiva osa video keskelt.

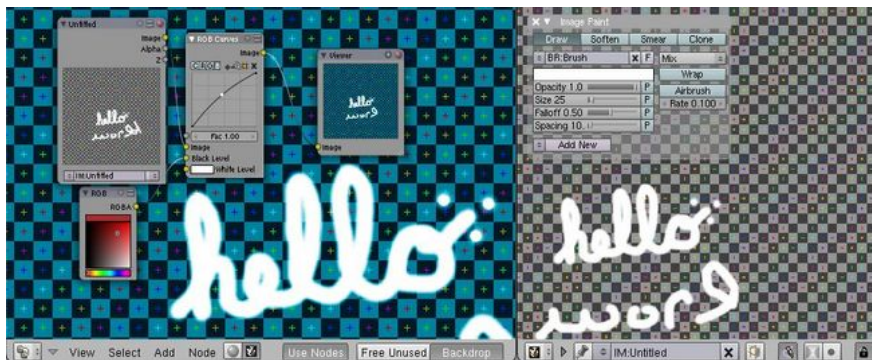
Nupp Frames (kaadreid) määrab, mitu kaadrit sa soovid laadida. Näiteks kui sa soovid laadida 2 sekundi pikkuse jupi ning su stseeni kaadrisagedus on 30 kaadrit sekundis, kirjuta sinna 60.

Start Frame (alguskaader) määrab, millises kaadris hakkab video sinu kompositsioonis mängima. Näiteks kui sa soovid, et sinu kompositsioonis hakkaks laetav video mängima kümnendal sekundil, kirjuta sinna 300 (kui kaadrisagedus on 30).

Nupp Offset (nihe) määrab, milline on esimene kaader, mida videost laetakse. Näiteks kui sa soovid, et video esimene laetav kaader oleks 20., kirjuta sinna 20.

Et panna video korduma, lülita sisse nupp Cyclic (korduv). Näiteks kui sul on video pöörlevast ventilaatorist ning ühe tiiru teeb ta 30 kaadri jooksul, võid sa need 30 kaadrit panna korduma, et ventilaator pidevalt pöörleks. Et näha sõlme pildiaknas parasjagu laetavat kaadrit, lülita sisse nupp Auto-Refresh (uuenda automaatselt).

Genereeritud pildid



Sõlmede kasutamine UV/pildiredaktoris avatud pildi manipuleerimiseks

Blender võimaldab [tekstuuride maalimist](#) läbi UV/pildiredaktori. Kõik aktiivsed tekstuurid on selleks läbi Blenderi kättesaadavad. Kui sünkroniseerimine on sisse lülitatud (lukuga ikoon UV/pildiredaktori päises), uuendatakse muudetud pildifail kohe kõigis kohtades, kus ta kasutuses on. Näiteks kui pilt on mõne pildisõlme (*Image*) sisendfailiks. Näites on kujutatud pildile maalimine UV/pildiredaktori aknas ning nagu näha, kajastuvad kõik muutused kohe ka komposiitoris. Uue pildi loomiseks vali redaktoris *Image->New* (pilt -> uus). Tekstuuride maalimise kohta leiad rohkem infot manuaali vastavast peatükist.

Vasakpoolses aknas on näha, et komposiitoris on sisendiks pildisõlm (*Image*) ja laetud on parasjagu maalitav "Untitled"-nimeline pilt. Pane tähele, et sisendfaili tüübiks on sõlmes märgitud *Generated*. Joonistatud pilt on teise sõlme abil värvitud ning tulemus on kuvatud sõlmeredaktori akna taustana.

Sedasi saad korraka komposiitida ning maalida. Taoline lähenemine on kasulik näiteks maskide loomisel. Kõik muutused on dünaamilised ning sa näed maalimise tulemusi kohe ka komposiitoris.

Märkused

Kaadrid laetakse videost üks-ühele: kui laetava failijada või video kiirus on originaalis erinev, kui stseeni seadetes määratud, mängib video kiiremini või aeglasemalt. Blender ei muuda sisendi kaadrisagedust, et erinevust kompenseerida. Kasuta [videoredaktori](#), et muuta video kiirus sobivaks, kui see peaks vajalik olema. Üks-ühele failide laadimist saad kasutada aegluubis võtete jaoks. Selleks salvesta video näiteks kaadrisagedusega 60 kaadrit sekundis ning lae ta stseeni, kus kaadrisagedus on 30. Tulemuseks on kaks korda aeglasemalt mängiv video. Vastupidise efekti saavutamiseks salvesta laetav video väiksema kaadrisagedusega.

AVI (*Audio Video Interlaced*) videofailid võivad sisaldada erineval viisil kokku pakitud videot. Video kokkupakkimise viisi (algoritmi) kutsutakse "koodeki"iks (*codec* ehk *enCOder-DECoder* ehk kokku-lahti pakkija). Kindla koodekiga salvestatud videofaili avamiseks peab see olema arvutisse installeeritud ning Blenderile kättesaadav. Vastasel korral kuvatakse veateade. Veateade **FFMPEG or unsupported video format** tähendab, et video pakkimiseks kasutatud koodek pole kättesaadav. Uuri, millise koodekiga on video pakitud ning katsu see endale installeerida. Koodeki tüübi ja info kuvamiseks videofailist on mitmeid vabavaralisi programme, näiteks VirtualDub. Erinevaid koodekeid loovad nii videoseadmete tootjad, tarkvarafirmad kui ka erinevad organisatsioonid. Näiteks Microsoft, Divx, Xvid, Cineform, Apple jt.

Videote lõikamine ja üleminekud

Eelpool kirjeldatud video sisendkaadrite valiku seadete ning segamissõlmede abiga saad luua erinevaid videoklippide üleminekuid. Üleminekuid on mitut erinevat tüüpi:

- Lõige - üks klipp lõpeb ja teine algab, üleminek on järsk
- Sisse sulatamine (*Fade In*) - klipp algab ühevärvilisena (enamasti musta või valgena) ja muutub järk-järgult tavaliseks
- Välja sulatamine (*Fade Out*) - klipp algab tavalisena ja muutub järk-järgult ühevärviliseks (tavaliselt mustaks või valgeks)
- Segunemine (*Mix*) - üks klipp sulab teiseks
- Värvi kastmine (*Dip To*) - üks klipp sulab alguses mustaks ja siis teiseks klipiks
- Liikumiseefektid (*Bumps and Wipes*) - üks klipp lükkab teise kaadrist välja

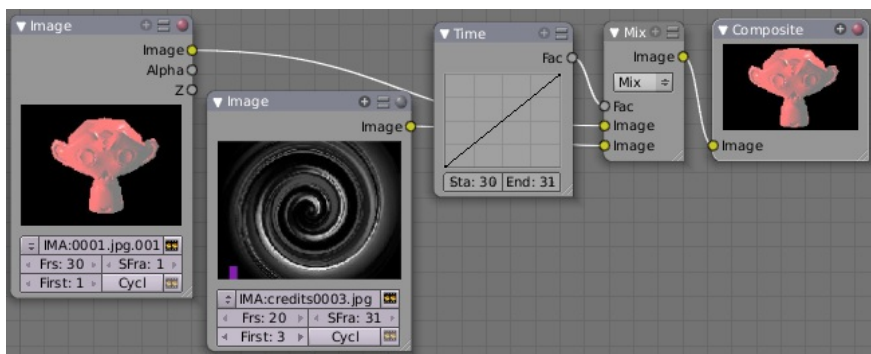
Lõike ülemineku loomine sõlmede abil

Allolevas näites on meil sisenditena kaks klippi, mida me soovime ühendada.

- Magic Monkey - failijada 0001.png kuni 0030.png
- Credits - tiitrid, failid credits0001.png kuni credits0030.png

Monteerija otsustas, et tiitrite esimesed kaks kaadrit ning viimased kaheksa võib välja lõigata ning jätta alles vaid keskmised 20. Näite pildil pole seda näha, aga tähtis on, et me määrasime renderdamise seadetest uuele ühendatud klipile mõistetava nime

(näiteks "Monkey-Credits-") ning seaksime animatsiooni pikkuseks 50 kaadrit (30 ahvi klipist ja 20 tiitritest). Pane tähele ajasõlme (*Time*), mis määrab segamissõlme (*Mix*) jaoks, millisest kaadrist alates tuleb teha sisendpildi vahetus.



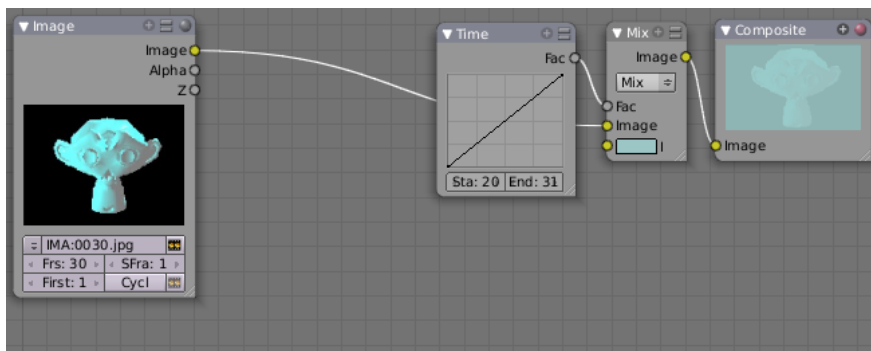
Lõike ülemineku loomine sõlmedega

Loodud sõlmekaardi renderdades saame tulemuseks kokku ühendatud klipid. Kui sa määrasid salvestatavaks formaadiks pildiformaadi, näiteks PNG, salvestab Blender kokku 50 faili, nimedega "Monkey-Credits-0001.png" kuni "Monkey-Credits-0050.png". Kui sa määrasid formaadiks video, näiteks AVI-JPEG, salvestab Blender kõik renderdatud kaadrid üheks videoks, antud juhul "Monkey-Credits-.avi" nimeliseks.

Kasuta lõikamisega üleminekut klippide kõige tavalisema üleminekuna või ka kiirete stseenide monteerimiseks, kus on palju tegevust ja pinget. Proovi lõikamisel jälgida ühendatavate klippide sisu (näiteks kaamera liikumist), et nad kenasti kokku sobiksid. Katsu vältida tumedast stseenist heledasse lõikamist, kuna see on silmi ärritav. Lõikamisega saad tekitada ka emotsionaalseid kontraste, näiteks vastandades kiiret liikumist aeglasele.

Sisse sulatamine

Eelmises näites selgitasime, kuidas ühendada kahte klippi lõikamise abil. Sisse või välja sulatamine on põhimõtteliselt samane, lihtsalt üks klipp on ühtlane värv (näiteks must või valge) ning ajasõlme üleminekuvaeg on pikem kui üks kaader. Alloleval pildil on kujutatud sulatamine sinakasrohelisteks värviks:



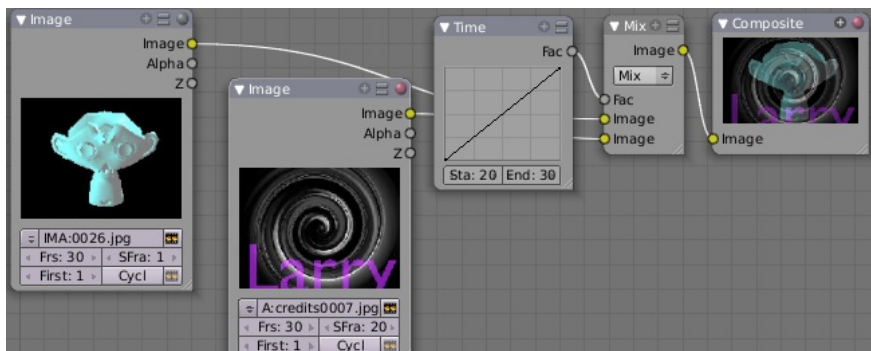
Sulatamine sõlmede abil

Sinine värv on siin valitud pärdiku värvi järgi, kuid sa võid kasutada ükskõik millist värvi. Pildil on kujutatud kaader 30, kui sulatamine on peaaegu lõpule jõudnud.

Sissesulutamise efekti loomiseks tuleb segamissõlmes (*Mix*) lihtsalt sisendid omavahel ära vahetada.

Sulatamine ühest klipist teise

Ühe klipi teiseks sulatamiseks saab kasutada täpselt samasugust sõlmekaarti nagu eelmises näites. Pildil on näha kaader 25, kui sulatamine on parasjagu pooleli. Üks klipp muutub aeglaselt läbipaistvamaks ning tema asemele ilmub teine klipp.



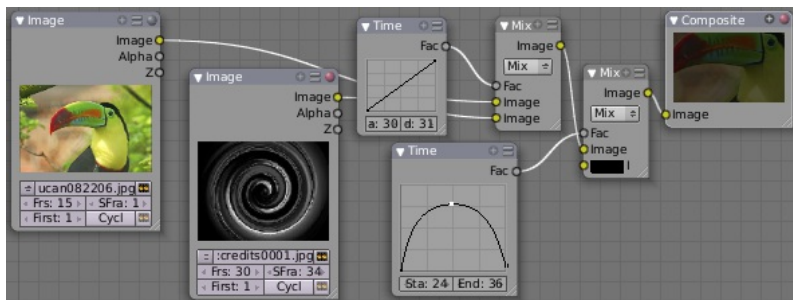
Sulatamine ühest klipist teise

Kasuta seda efekti näiteks kahe samase stseeni üleminekul või näitamaks, et vahepeal on möödunud palju aega. Näiteks: ühes klipis näeme me pangaröövlit mööda tänavat kõndimas, kaamera filmib röövli jalgu. Teises klipis näeme samasugusest rakursist filmitud politseiniku jalgu. Ühte klippi teiseks sulatamisega loome tunde, et politseinik jälitab pangaröövlit.

Värvi kastmine

Värvi kastmine on efekt, mille puhul sulab pilt esimesest klipist miniks värviks ning seejärel teiseks klipiks. Selle efekti loomiseks

peame kasutama eelnevates lõikamise ja sulatamise näidetes loodut:



Värvi kastmise efekt

Juuresolevas näites sulatame me linnu pildi mustaks ning musta seejärel keerise pildiks:

- Kaadri hoidmine lõpus: kuigi meil on vaid 15 kaadrit linnu pilti ja üleminek ei toimu enne kaadrit 30. Blender väljastab kaadrite lõppemisel viimast kaadrit, pikendades sedasi videot sobival määral.
- Kaadri hoidmine alguses: kuigi keerise pildijada hakkab mängima alles kaadris 34, annab Blender vajalikud kaadrid ka enne seda, väljastades selleks kõige esimest kaadrit. Sel viisil hoolitseb Blender, et pilt ootamatult otsa ei lõpeks.
- Osaline kastmine: pane joonisel tähele teist ajasõlme, mis teeb kastmise osaliselt. Erinevalt esimesest ei sulatata siin klippi vahepeal täielikult mustaks, vaid ainult umbes 75% jagu. Kasuta seda, kui sa tahad vaatajat sujuvamalt ühest stseenist teise viia.

Mitu sisendit

Ülaltoodud näiteks kasutasime vaid kahte sisendit, kuid sõlmi *Input* (sisend), *Time* (aeg) ja *Mix* (segamine) kopeerides ning sobivalt ühendades saad korraga kokku sulatada ka mitu erinevat pilti või videot.

Tekstuurisõlm (Texture)

Panel: [Sõlmeredaktor](#) → [Node Composition](#) (komposiitimine)

Menu: ⇧ ShiftA → [Input \(sisend\)](#) → *Texture* (tekstuur)



Tekstuurisõlm

Texture

Tekstuurisõlm (Texture) annab võimaluse kasutada komposiitmisel Blenderi 3D-tekstuure.

Kasutada saab tekstuure, mis on parasjagu avatud failist kättesaadavad.

Märkus

Uuri ka Blenderi kataloogi- ja linkimissüsteemi toimimist - selle abil saad importida tekstuure ka teistest .blend-failidest.

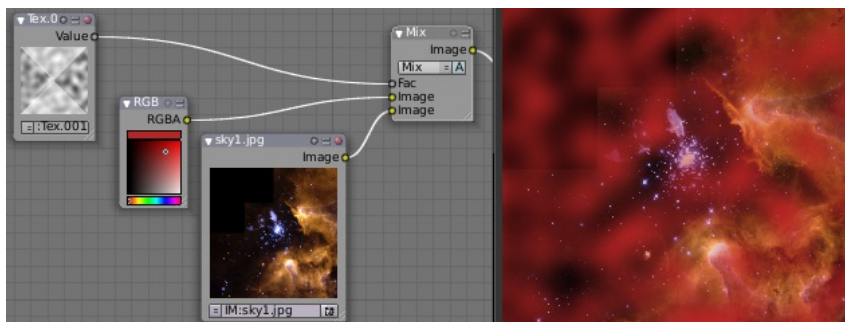
Märkus

Tekstuure on võimalik komposiitoris vaid kuvada, neid sealtkaudu muuta pole võimalik. Tekstuurisõlme kasutamiseks pead sa teksuuri kõigepealt valmis tegema (kasuta selleks tekstuuripaneeli või UV/pildiredaktorit) ning seejärel saad ta sõlme kaudu komposiitorisse laadida.

Sa saad muuta tekstuuri nihke (Offset) ja suuruse (Scale) seadeid, et määrata, milline osa tekstuurist laetakse. Mõlemad seaded nõuavad vektorsisendit (XYZ), kuna tegemist on 3D-tekstuuridega ja neid saab nihutada ning suurendada/vähendada kolme telje suunas.

Tekstuurisõlmed suudavad genereerida nii mustvalge pildi ehk väärtuspinna (Value - ära aja seda segamini läbipaistvusega - kui ka värvilise pildi (Color).

Näide



Oletame, et me soovime simuleerida kena punakat plasmagaasi, et täita meie muidu staatiline kosmosepilt liikumisega. Kosmose pilt on pärit Hubble'i teleskoobist ning kujutab Orioni vööd. Lisame sellele pilve tüüpi tekstuuri (*Cloud*), et muuta pilt dünaamilisemaks.

Väärtuse genereerimise sõlm (*Value*)

Panel: [Sõlmeredaktor](#) → [Node Composition \(komposiitimine\)](#)

Menu: ⇧ ShiftA → [Input \(sisend\)](#) → *Value* (väärtus)

Väärtussõlmel (*Value*) pole sisendeid ning tema väljundiks on vaid üks number vahemikus 0.00 kuni 1.00. Väljastatava numbri saad toksida vastavasse sisestuskasti.


Kasuta seda sõlme, et anda sisendit sõlmedele, mis nõuavad mõne sisendina väärtust.

RGB sõlm (*RGB*)

Panel: [Sõlmeredaktor](#) → [Node Composition \(komposiitimine\)](#)

Menu: ⇧ ShiftA → [Input \(sisend\)](#) → *RGB*

RGB sõlmel sisendid puuduvad. Ta väljastab värvi, mille saad määrata sõlme seadete kaudu. Juuresoleval pildil on valitud sinakas toon.

Värvi tooni ja küllastuse muutmiseks klõpsa LMB  värvipaletil sobivas kohas. Valitud värvipositsioon kuvatakse pisikese ringikesega. Värvi heleduse muutmiseks klõpsa paleti kõrval olevat heledusribal sobivas kohas.

Näide

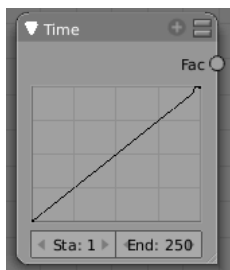


Siin on meil kena tekst, kuid see peaks olema sinakas, mitte valge. Seega korrutame me selle igava mustvalge pildi sinaka värviga läbi, et saada soovitud tulemus.

Ajasõlm (*Time*)

Panel: [Sõlmeredaktor](#) → [Node Composition \(komposiitimine\)](#)

Menu: ⇧ ShiftA → [Input \(sisend\)](#) → *Time* (aeg)



Ajasõlm (*Time*)

Ajasõlm genereerib väärtuse vahemikus 0.00 kuni 1.00 vastavalt stseeni aktiivsele kaadriale.

Nupud Start (algus) ja End (lõpp) määravad, millises kaadrite vahemikus peaks väärtuse muutmine aset leidma. See on graafikul X-telg. Väljundväärtuse saame Y-teljelt ning kõvera kuju muutes saame sisendkaadritele panna vastama erinevaid väärtuseid. Näites toimub väärtuse muutus 250 kaadri jooksul ning kuna ülekandekõver on sirge, siis kaadris 125 oleks väljundväärtuseks 0.5 ning kaadris 187 0.75.

Märkus väljundväärtuste kohta

[Map Value](#) (väärtuse teisendamise) sõlme abil saad sa skaleerida väljundväärtused sulle vajalikku vahemikku, näiteks kas või -5.0 kuni 15.0. Mõnel juhul võib ajasõlm väljastada ka arvu, mis on suurem kui 1.0 või väiksem kui 0.0. Vigaste vahemike vältimiseks

kasuta *Map Value* sõlme *Min/Max* seadeid, et lubamatud väärtused maha lõigata.

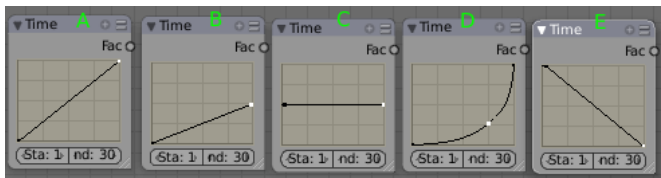
Ajasõlm võimaldab aega ka "tagurpidi" keerata, selleks peab sõlme alguskaader (*Start*) olema suurem kui lõppkaader (*End*). Tulemuseks on põhimõtteliselt X-telje suhtes peegeldatud kõver. Hoiatus: aja pööramise kasutamine võib hiljem kergesti ununeda ja põhjustada palju segadust, kui sõlm produtseerib näiliselt valesid väärtuseid.

Aeg on suhteline

Blenderis mõõdetakse aega kaadrites. Video või animatsiooni reaalne pikkus sõltub sellest, mitu kaadrit sekundis me seda mängime. Kaadrisageduse (inglisekeelne lühend *FPS* ehk *Frames Per Second*) saab määrata renderdamise seadetest ([stseenikontekstis](#) F10). Tavaliselt jääb kaadrisagedus vahemikku 5 kaadrit sekundis (0.2 fps, näiteks slaidiprogrammid) kuni 30 fps (USAs kasutatav NTSC video).

Näiteid ajasõlmedest

Alloleval pildil on kujutatud erinevat tüüpi ajakõveraid:



A) Lineaarne B) Aeglustuv C) Seisev D) Kiirenev E) Tagurpidine

Ajasõlme kasutatakse tavaliselt erinevate sulatamiste puhul, kus genereeritav väärtus määrab ära kahe pildi segamise vahetunde. Teine mõistlik kasutus on näiteks udustamise efektid, kus saab pilti sujuvalt udusemaks või selgemaks muuta.

Udusta näiteks ajasõlmega esiplaani kihti ning teise, pööratud ajasõlmega tagaplaani. See loob mulje, nagu fookuseeriks keegi virtuaalset kaamerat.

Näited ja soovitused

Sisendsõlmed on vahest kõige olulisemad sõlmed, kuna ilma nendeta ei oleks meil ka pilte, mida muuta või kokku kombineerida. Nende abil saad luua erinevaid efekte, kombineerides näiteks omaloodud animatsiooni või filmitud videot värvide, mustrite või muude elementidega. Lase fantaasial lennata ning järgmine kord, kui sul on vaja näiteks järk-järgult udusemaks muutuvat klaasi või anda edasi muljet, et keegi just ärkas ja maailm on nii valge, nii valge, siis nuputa, kuidas selle efekti loomiseks erinevaid sõlmesid võiks kombineerida.

Nagu ikka, võib samalaadse tulemuse saavutamiseks olla mitmeid erinevaid võimalusi. Näiteks Blenderi varasemates versioonides kasutati slaidishow' loomiseks hunnikut pildiga tekstuure ning kaarti (*Plane*), mis täitis kaamera vaatevälja. Piltide vahetamiseks tuli animeerida iga tekstuuri mõju kaardi materjalile ning lõpuks kõik see läbi kaamera välja renderdada. Oli see alles jupsimine! Slaidide järjekorra muutmine oli väga tülikas ning nõudis materjali võtmete ümberpaigutamist. Sõlmekaardi abil saad sama asja palju kiiremini valmis teha. Kasuta näiteks pildi (*Image*) sisendsõlmi ja ajasõlmi (*Time*) ning sulata pilte sobivalt üksteiseks. Veelgi lihtsam lahendus oleks aga laadida slaidid sisse pildijadana ning muuta animatsiooni kaadrisagedust.

Komposiitimise väljundsõlmed (*Output*)

Et näha, milline on meie sõlmekaardi tulemus, või selleks, et tulemust pildina salvestada, on meil vaja väljundsõlmi. Väljundsõlm ise ühtegi operatsiooni ei soorita, ta lihtsalt kuvab meile sõlmekaardi tulemuse.



Korraga saab aktiivne olla vaid üks kuvamissõlm (*Viewer*) ja üks komposiitkuva sõlm (*Composite*), millele viitab sõlme päises olev punase keraga ikoon. Kuvamissõlmel klõpsamine muudab vastava sõlme aktiivseks. Komposiitkuva sõlmedest on aktiivne alati esimesena lisatu ning pealegi pole rohkem kui ühte vaja kasutada.

Kuvamissõlm (*Viewer*)



Kuvamissõlm
(Viewer)


Kuvamissõlm Viewer võimaldab näha vahetulemust sõlmekaardi vabalt valitud kohast. Ühenda tema sisendisse ükskõik millise teise sõlme väljund, et seda väljundit näha.

Klõpsa LMB  sõlme pisipildil, et seda uuendada (kui uuendamine ei toimunud automaatselt). Kuvamissõlmi võid lisada nii palju kui soovid. Võimalik on igale sõlmele automaatselt kuvamissõlm lisada vajutades selle peal ⇧ Shift LMB .

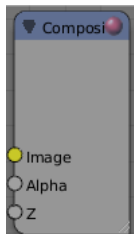
UV/pildiredaktori kasutamine kuvamissõlmega

Kuvamissõlm (Viewer) võimaldab näidata pilti ka pildiredaktori aknas. Selleks tuleb redaktori sisendpildiks valida IM:Viewer Node (kuvamissõlm). Kuvatakse pilt, mis on parasjagu aktiivse kuvamissõlme sisendiks.

Pildi salvestamiseks võid kasutada redaktori tavapäraselt Image->Save As (pilt -> salvesta kui) käsku.

UV/pildiredaktori abil saad lihtsa vaevaga kuvada ka pildi alfakanalit või näiteks Z-kanalit. Selleks peavad vastavad kanalid olema ühendatud kuvamissõlme sisenditesse. Klõpsates LMB  pildil, on sul võimalik näha vastava piksli värvi ja läbipaistvuse väärtuseid.

Komposiitkuva (*Composite*)



Komposiitkuva
sõlm
Composite

Komposiitsõlme ühendamise saab töödeldud pildi renderdajasse tagasi. Selle ühendamata jätmise, kui komposiitimise kasutamine on sisse lülitatud, annab tulemuseks tühja pildi! Seda sõlme värskendatakse pärast igat renderdust ning siis, kui sa muudad sõlmekaardis mõne sõlme parameetreid.

Komposiitsõlme sisendisse saad sa ühendada nii RGBA pildi, eraldi alfakanali kui ka Z-kanali. Sõlmekaardis peaks olema vaid üks komposiitsõlm, kuna vaid esimene lisatud sõlm on aktiivne. Lisaks peab renderdamise seadetest olema sisse lülitatud komposiitimise lubamine (valik Compositing). Vastasel korral ei saa sa lihtsalt pilti ette!

Komposiitpildi salvestamine

Nupp Render Image (renderda pilt) renderdab ühe (stseeni aktiivse) kaadri või sõlmekaardi väljundi. Pildi salvestamiseks kasuta kiirklahvi F3 või menüükäsku File->Save Image (fail -> salvesta pilt). Pilt salvestatakse renderduspaneelil määratud seadetega.

Pildijada (animatsiooni) salvestamiseks klõpsa nupul Animation. Ära unusta enne määramast animatsiooni algus- ja lõppkaadrit! Kui sa soovid renderdatud pilte hiljem komposiitmisel kasutada (näiteks paigutada renderdatud elementi mõne teise animatsiooni sisse), siis kasuta jada salvestamisel formaati, mis toetab alfakanalit (näiteks PNG). Kui sa tahad hiljem lisada sügavusdu efekte või kasutada erinevaid renderduskäike, kasuta formaati, mis vastavaid kanaleid toetab. Sobiv valik on näiteks EXR.

Animatsiooni salvestamisel videofailiks (kõik kaadrid ühes failis koos) vali formaadiks näiteks Xvid, H.264 või MPEG.

Võrdluskuva sõlm (*SplitViewer*)



Võrdluskuva sõlm
SplitViewer

Võrdluskuva (SplitViewer) sõlm võtab kaks sisendpilti ning näitab kummastki ühte poolt, asetades nad kõrvuti. Ülemine ühendus on parem pool ja alumine vasak pool. Kasuta seda sõlme piltide kõrvuti võrdlemiseks, näiteks väikeste erinevuste otsimiseks või tonaalsuse võrdlemiseks värvikorrekatuuril. Võid kasutada seda ka näiteks klippide monteerimisel, et võrrelda ühe klipi viimast ning teise esimest kaadrit.

Salvestussõlm (*File Output*)



Salvestussõlm *File Output*

Salvestussõlm võimaldab salvestada sõlmekaardi väljundi pildifailiks või pildijadaks. Määrata saab kaadrite vahemiku, mida salvestada, pildi formaadi ning nime. Jada renderdamisel lisatakse failinimele kaadri järjekorranumber. Sõltuvalt valitud formaadist võidakse sulle kuvada erinevaid kvaliteedi ja pakkimise seadeid.

Sõlm võimaldab salvestada ka Z-kanalit, kuid selleks peab pildi formaat seda toetama. Sobiv valik selleks on OpenEXRi formaat.

Kuidas siis kaadrit selle sõlme abil salvestada? See on väga lihtne. Selleks tuleb vaid klõpsata kaadri renderdamise nupul (menüüst *Render -> Image* (renderda -> pilt) või renderdamise paneelil nuppu *Image*). Animatsiooni renderdamiseks vali Menüüst *Render -> Animation* (renderda -> animatsioon) või klõpsa nupul *Animation*.

Selle sõlmega ei pea sa enam meeles pidama, et pildifaili salvestada, sõlm salvestab pildi alati pärast renderdamist. Kuna sõlme saab ühendada sõlmekaardis suvalisse kohta, võimaldab see automaatselt salvestada ka erinevaid vahetulemusi. Kas pole mitte kaval?

Faili kataloogiviidad

Kõigis kohtades, kus on vaja viidata laetavale või salvestatavale failile, võid kasutada kataloogitee alguses // märke, et viidata .blend-faili kodukataloogile. Samuti saad kasutada kahte punkti .., et liikuda kataloogipuus üks samm tagasi. Näiteks faili viide ../pilt.jpg viitab .jpg-failile mis asub .blend-faili kodukataloogi sisaldavas kataloogis, st üks samm kataloogipuus tagasi.

Statistikasõlm (*Levels Node*)

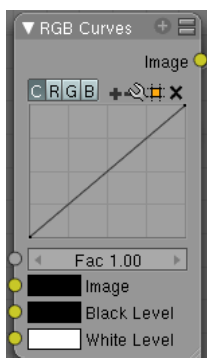
Statistikasõlm arvutab sisendpildilt erinevaid parameetreid ning väljastab saadud tulemuse. Arvutuses kasutatakse vastavalt valikule sisendi kombineeritud RGB (C), punase (R), roheline (G), sinise (B) või heleduse (L) kanalit.

Väljundiks võib olla kanali keskväärtus (Mean) või standardhälve (Std Dev), mis mõõdab piksliväärtuste hajusust.

Komposiitimise värvisõlmed (*Color Nodes*)

Need sõlmed tegelevad pildi värvinformatsiooniga. Võimalik on muuta värvide intensiivsust, pildi kontrastsust ja segada kaks pilti kokku värvinfo, läbipaistvuse ning sügavuskanali Z alusel.

RGB-kõverate sõlm (*RGB Curve*)

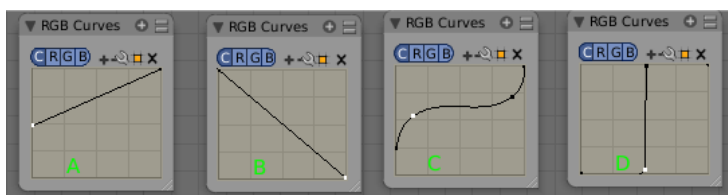


RGB-kõverate sõlm
RGB Curves

See sõlm võimaldab määrata igale värvikanalile (RGB) või komposiitpildile (C) Bezier' ülekandekõvera, mis määrab sisendväärtuste (x-teljel) alusel väljundväärtused (y-teljel). Vaikimisi on ülekandefunktsiooniks ühtlase tõusuga sirge, nii et sisendväärtus 0.5 x-teljel annab väljundiks 0.5 y-teljel. Uue kontrollpunkti lisamiseks klõpsa kõveral ja lohista uus punkt sobivasse kohta. Kasuta kiirklahvi X, et kustutada valitud (valge) punkt.

Klõpsamine värvikomponendi tähisel C R G B näitab vastava komponendi ülekandekõverat. Näiteks ülekandekõvera lamedamaks tegemine (lohistades vasakpoolset punkti y-teljel üles) surub sisendväärtused kokku väiksemasse vahemikku ja tõstab pildi üldist heledust. Sel viisil saame me ähmaseid detaile nähtavamaks muuta, aga vähendame ühtlasi ka pildi kontrastsust. Ülekandekõverat saab määrata igale värvikanalile eraldi, näiteks võime luua kõvera, mis väljastab punases värvikanalis tooni vaid siis, kui seda on suures koguses.

Näited mõnedest enamlevinud ülekandekõvera kujudest, millega saavutada erinevaid efekte:



A) Heledam B) Negatiiv C) Kontrasti vähendamine D) Posteriseerimine

Valikud

Fac (kordaja)

Kui palju peaks sõlm mõjutama väljundväärtusi

Black Level (musta tase)

Määrab sisendväärtuse, mis saab väljundis väärtuseks 0 ehk muudetakse mustaks. Vaikeväärtus on 0 ehk must ja ei muuda pilti.

White Level (valge tase)

Määrab sisendväärtuse, mis väljundis saab väärtuseks 1 ehk muudetakse valgeks. Vaikeväärtus on 1 ehk valge ja ei muuda pilti.

Tasemed töötavad samalaadselt tasemete funktsionaalsusega pildiaknas. Sisendväärtused skaleeritakse lineaarselt vahemikust 0...1 vahemikku musta tase...valge tase.

Tasemete väärtuse määramiseks võid klõpsata vasaku hiireklahviga värvivalijas või ühendada sõlmega mõne RGBA sisendi.

Et muuta ainult heledust või kontrasti ja mitte toone, peavad tasemed olema kõigis värvikanalites sama väärtusega ehk halltoonid. See on samaväärne sellega, kui määrad lineaarse ülekandefunktsiooni C-kanalile.

Kui sa sead mõne taseme mitte halltooniks, vaid värviks, mille küllastus on suurem kui 0, siis muutuvad ka väljundpildi värvid vastavalt, võimaldades teha lihtsat värvikorrektsiooni või efekte. See on samaväärne sellega, kui määrad lineaarsed ülekandefunktsioonid värvikanalites R, G ja B.

Näited

Värvikorrektuur kõverate abil



Värvikorrektuur kõverate abil

Antud näites on pilt liiga punane, seega lisame me RGB-sõlme ja vähendame punase kanali väärtusi umbes poole võrra.

Lisame ka vahepunkti kõvera keskele, et muuta see logaritmilise kõvera sarnaseks. Selline kõver vähendab punase värvi ülekuüllastust. Loe edasi, et tutvuda näidetega tumendamise ja kontrasti muutmise kõverate kohta.

Värvikorrektuur musta/valge tasemetega



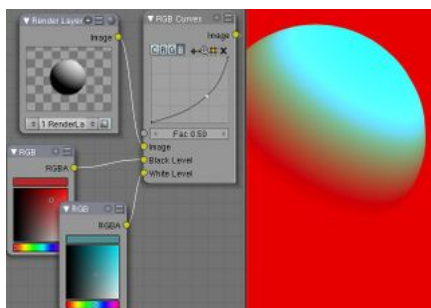
Värvikorrektuur musta/valge tasemetega

RGB-kõverate käsitsi määramine värvikorreksiooniks võib olla keerukas. Teine variant värvikorrektuuriks on kasutada selle asemel musta ja valge tasemeid.

Antud näites on valge tasemeks määratud heleda liiva värv tagaplaanilt ning musta tasemeks värv kala silma keskelt. Efektivsemaks tööks on kasulik sõlmekaarti lisada pildi kuvamise sõlm ja määrata selle sisendiks originaalpilt. Sel viisil saab tasemete värvipipetiga valida sobivad tasemed otse sisendpildilt (soovikorral seda suurendades), mitte juba osaliselt korrigeeritud väljundilt. Tulemust on võimalik täpsemalt korrigeerida RGB-kõverate abil sarnaselt eelmisele näitele.

Komposiitpildi kõverat (kanal C) saab kasutada musta ja valge tasemete muutmise kaasnähuna tõusnud kontrasti korrigeerimiseks.

Efektid



Värviteisendused

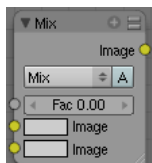
Kõveraid ja musta/valge tasemeid saab kasutada ka pildi värvide täielikuks muutmiseks.

Pane tähele, et punase värvi määramine musta tasemeks ning sinise värvi määramine valge tasemeks ei asenda lihtsalt musta punasega ega valget sinisega. Tasemete määramine muudab värvikanalite ülekandefunktsioone, mitte ei asenda värve, kuid sõltuvalt seadetest võib saavutada ka kirjeldatud värviasenduse efekti.

(Tegelikult pööratakse antud juhul punane värvikanal vastupidiseks, roheline kanal nullitakse täielikult ning sinine jääb muutmata.)

Seetõttu võib musta/valge tasemete või RGB-kõverate muutmise tulemus olla raskesti ennustatav, kuid sellegipoolest olla kasulik ja huvitav.

Segamissõlm (*Mix*)



See sõlm segab baaspildi (ühendatud ülemisse sisendpessa) teise pildiga (alumine pesa), töödeldes piltide üksteisele vastavaid piksleid. Väljundi loomiseks kasutatava funktsiooni saab valida rippmenüüst. Väljundpildi suurus (lahutus) on määratud baaspildi suurusega. Lisaks värvikanalitele segatakse ka alfa (läbipaistvus) ja Z-kanalid.

Kuusteist erinevat segamifunktsiooni on järgnevad:

Mix (segamine)

Baaspilt B kaetakse pildiga A läbipaistvuse (alfa) väärtuste alusel.

Add (liitmine)

Piltide A ja B piksliväärtused liidetakse. Seade *Fac* määrab, kui palju pilt A väljundit mõjutab. Tulemus on heledam pilt.

See on "vastand" lahutamise funktsioonile (Subtract).

Subtract (lahutamine)

Pildi A piksliväärtused lahutatakse B piksliväärtustest. Tulemus on tumedam pilt.

See on "vastand" liitmise funktsioonile (Add).

Multiply (korrutamine)

Piltide A ja B piksliväärtused korrutatakse. Tulemuseks on tumedam pilt, välja arvatud juhul, kui ühe pildi pikslid on valged.

Täiesti valge pilt A või B sisendit ei muuda. Täiesti must pilt A või B annab tulemuseks musta pildi.

See on "vastand" funktsioonile Screen (varjamine).

Screen (varjamine)

Pikslite pöördväärtuste korrutise pöördväärtus $1-(1-A)*(1-B)$. Tulemuseks on üldjuhul heledam pilt kui kumbki sisendpildidest

(välja arvatud juhul, kui üks neist on must). Kui A või B on täiesti must, siis teine pilt ei muutu, täiesti valge annab aga tulemuseks valge pildi.

See on "vastand" funktsioonile Multiply (korrutamine).

Overlay (katmine)

Kombinatsioon funktsioonidest Screen ja Multiply, tulemus sõltub pildi B piksliväärtustest.

Divide (jagamine)

Baaspildi B piksel (ülemine sisend) jagatakse pildi A piksliväärtusega: kui A on valge (= 1.0), siis on väljundiks B; mida tumedam on A, seda heledam on tulemus (jagamine väärtusega 0.5 on sama mis korrutamine väärtusega 2.0); kui A on must (= 0.0, jagamine nulliga on võimatu!), siis B piksliväärtust ei muudeta ja tulemuseks on B.

Difference (erinevus)

Piltide A ja B piksliväärtuste vahe absoluutväärtus. Tulemuseks on erinevus kahe pildi vahel - must tähendab, et erinevust ei ole, valge aga maksimaalset võimalikku erinevust. Tulemus näeb enamasti välja suhteliselt veider. Seda funktsiooni saab kasutada pildi B negatiivi pööramiseks või näiteks kahe pildi erinevuste leidmiseks (tulemuseks on must pilt, kui nad on võrdsed).

Darken (tumendamine)

Piltide A ja B piksliväärtustest valitakse väiksem. Täiesti valge pilt ühe sisendina tulemust ei muuda, täiesti must pilt annab tulemuseks musta.

Lighten (helendamine)

Piltide A ja B piksliväärtustest valitakse suurem. Täiesti must pilt ühe sisendina tulemust ei muuda, täiesti valge pilt annab tulemuseks valge.

Dodge (alasäri)

Tõstab pildi B heledust pildi A väärtuste alusel. Tulemuseks on heledam pilt nendel aladel, kus pilt A on heledam. Kasuta parameetrit "Fac", et määrata, kui palju pilt A tulemust mõjutab.

Burn (ülesäri)

Langetab pildi B heledust pildi A heleduse alusel. Tulemuseks on tumedam pilt, kuna A "põletatakse" B sisse. Rohkem valgust annab tumedama tulemuse, kuna tegemist on negatiivprotsessiga.

Color (värv)

Pildi B värv määratakse pildi A järgi, jättes heleduse muutmata. Kasuta seda pildi toonimiseks.

Value (väärtus)

Mõlema pildi RGB piksliväärtused konverteeritakse HSV (*Hue, Saturation, Value* ehk toon, küllastus ja heledus) väärtusteks. Tulemuse heleduseks saab piltide A ja B heleduste keskvärtus ning toon ja küllastus jäävad pildi B omad.

Saturation (küllastus)

Mõlema pildi RGB piksliväärtused konverteeritakse HSV väärtusteks. Tulemuse küllastuseks saab piltide A ja B küllastusväärtuste keskvärtus ning toon ja heledus jäävad pildi B omad.

Hue (toon)

Mõlema pildi RGB piksliväärtused konverteeritakse HSV väärtusteks. Tulemuse tooniks saab piltide A ja B toonide keskvärtus ning küllastus ja heledus jäävad pildi B omad.

Värvikanalid

Kindla värvi väljendamiseks on võimalik kasutada kahte erinevat kolmeteljelist süsteemi: RGB ja HSV. RGB tähistab punast, rohelist ja sinist (*Red, Green, Blue*) ning HSV tooni, küllastust ja heledust (*Hue, Saturation, Value*). Kummagi kolme parameetri järgi saab määrata kõiki võimalikke esitatavaid värve.

Alfa (läbipaistvus)

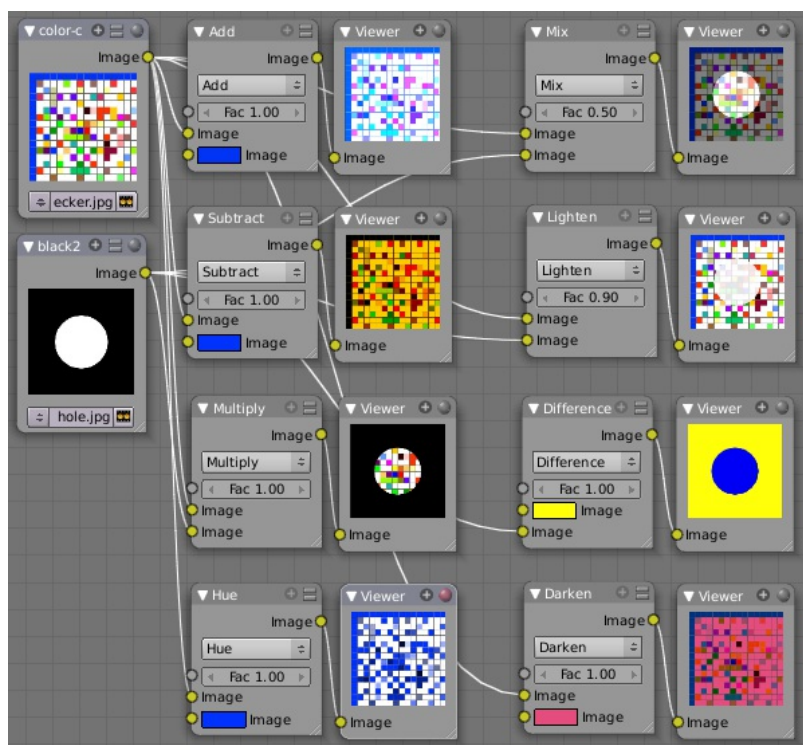
Klõpsa nupul Alpha (läbipaistvus) et panna segamissõlm (*Mix*) kasutama pildi B alfakanalit (läbipaistvuse väärtusi). Kui nupp on sisse lülitatud, on väljundpildi alfakanal mõjutatud mõlema sisendpildi alfakanalist. Muudel juhtudel (nupp on välja lülitatud ja heleroheline) segatakse väljundpildi värvid vastavalt A pildi alfakanalile. Väljundpildi alfakanalit ei muudeta, sellesse jäävad pildi B piksliväärtused.

Fac (kordaja)

Piltide A ja B segamise mõju määrab kordaja väli (Fac:). Kordaja 0.0 puhul ei kasutata pildi A väärtusi üldse, kordaja 1.0 puhul aga on mõju maksimaalne. Segamiskutsiooni (*Mix*) puhul annab kordaja väärtus 0.5 tulemuseks 50:50 segu kahest pildist, liitmiskutsiooni (*Add*) puhul aga tähendab kordaja 0.5, et pildi A mõju on vaid pool võimalikust.

Näited

Allpool on toodud näited mõnedest enamlevinud segamiskutsioonidest, kasutades värvilist ruudustikku ja mustvalget maski.

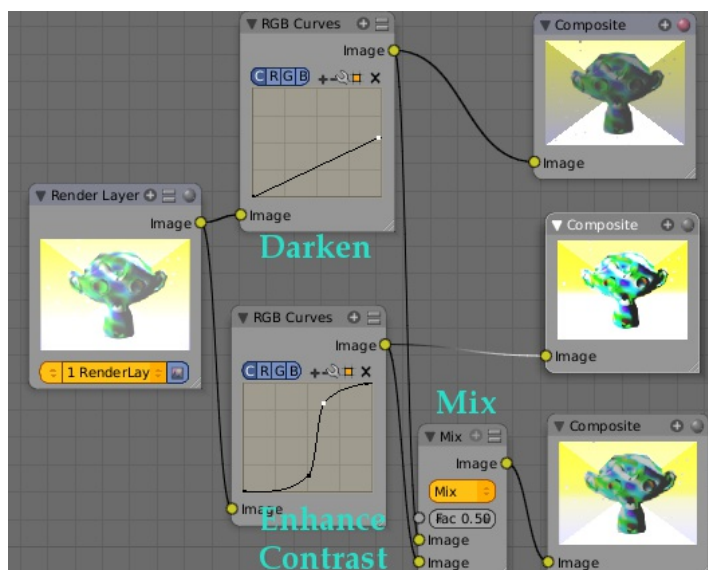


Mõned näpunäited ja seletused segamisfunktsioonide kasutamiseks:

- Add** (liitmine) - sinise lisamine sinisele värvile ei muuda, sinise lisamine punasele annab aga tulemuseks lilla. Valge värv sisaldab juba maksimaalselt sinist, seega on tulemuseks valge. Kasuta funktsiooni **Add** selleks, et nihutada pildi värve. Sinise lisamine muudab pildi toonid külmemaks.
- Subtract** (lahutamine): Sinise lahutamisel valgest jäävad alles punane ja roheline, mis annavad kokku kollase. Sinise lahutamisel lillast jääb alles punane. Kasuta seda, et vähendada pildi küllastustaset. Kollase (ehk punase ja rohelise) vähendamine muudab pildi sinakamaks.
- Multiply** (korrutamine): must (0.00) korrutatud ükskõik millise väärtusega annab tulemuseks musta. Valgega korrutamine (1.00) annab tulemuseks sama värvi. Kasuta seda mittevajalike pildi osade maskeerimiseks või mustvalge pildi toonimiseks.
- Hue** (toon): asendab pildi originaaltoonid teise pildi omadega või määratud tooniga.
- Mix** (segamine): kombineerib kaks pilti, andes tulemuseks nende keskmise.
- Lighten** (helendamine): pleegitab pilti. Kasuta seda koos maskiga, et muuta määratud alasid heledamaks.
- Difference** (erinevus): võimaldab koloreerida mustvalget pilti ning nihutada värve. Kollasest valge saamiseks on vaja sinist - erinevus puna-rohelise ja sinise vahel on maksimaalne ehk tulemuseks on valge. Kasuta seda funktsiooni, et võrrelda kahte väga sarnast pilti ja leida erinevused. Seda võib kasutada ka peidetud [vesimärgi](#) leidmiseks, kui sul on olemas vesimärgita originaal.
- Darken** (tumendamine) valib kahe pildi väärtustest tumedama, kasuta seda näiteks heledal taustal paikneva elemendi lisamiseks teise heleda pildi peale.

Kontrasti võimendamine segamisfunktsiooniga

Joonisel on kujutatud kahte RGB kõvera levinud kasutust: **Darken** ehk tumendamine ja **Contrast Enhancement** ehk kontrasti võimendamine. Näha on mõlema kõvera efekt eraldi ning kombineeritud efekt, kui nad on segatud 50:50 suhtes.



Näide tumendamise, kontrasti võimendamise ja segamise sõlmedest komposiitmisel.

Originaalpildil on pärdik kergelt üle valgustatud. Vea parandamiseks peab muutma pilti tumedamaks ja tõstma kontrastsust. Teised

pilditöötlusprogrammid võimaldavad tavaliselt muuta meid omadusi liuguritega, Blenderis aga on kasutaja määratud ülekandeköver, mille abil saab tulemust täpselt juhtida.

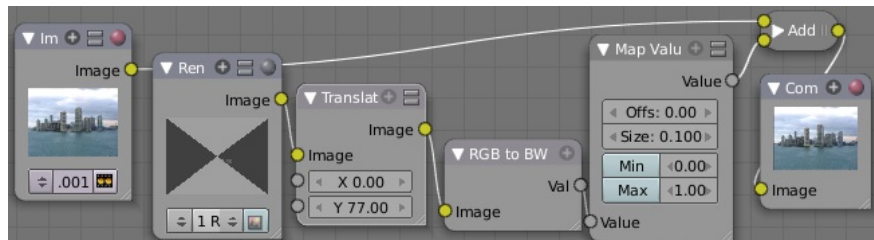
Ülemisel RGB-kõveral (tumendamine) on langetatud kõvera parempoolset otsa ning seega muudetakse heledad piksliväärtused tumedamaks. Kontrasti võimendamise kõvera iseloomulik 'S'-kuju tõstab kontrastsust heledusväärtuste keskmises osas ning surub kokku tumedad ja heledad toonid. Tulemuseks on kontrastsem pilt ning samas pole me vähendanud heledusväärtuste koguulatust (must jääb mustaks ja valge valgeks). Kõvera kuju määramiseks ja kontrollpunkti lisamiseks klõpsa kõveral sobivas kohas. Lohista kontrollpunkti ringi, et anda kõverale sobiv kuju. Segamissõlm (*Mix*) segab kaks RGB-kõvera väljundpilti ning tulemuseks on parandatud Blenderi pärdiku Suzanne'i pilt. Koledat pärdikut ei salli ju keegi.

Segamissõlme kasutamine pildi vesimärgistamiseks

Vanasti pressiti dokumentide vesimärgistamiseks märga paberimassi kujutis, et määrata, kes on autor ja kust dokument pärit on. Märk oli märgatav vaid õiges valguses ja õige nurga alt. Tegu võis olla ka esimese teadliku alateadvusele suunatud reklaamiga. Tänapäeval kasutatakse piltide vesimärgistamist autorluse kinnitamiseks, alateadvusele mõjuvaks reklaamiks ning ka pildimaterjali kasutuse jälgimiseks veebis. Blender pakub piltide vesimärgistamiseks ning vesimärgi olemasolu kindlakstegemiseks kõiki vajalikke tööriistu.

Vesimärgi lisamine pildile

Kõigepealt loo oma isiklik vesimärk. Võid kasutada näiteks oma nime, mõnda sõna, kujundit või pilti, mis pole laialt levinud. Neutraalhall kujutis töötab pakutud kodeerimismeetodiga kõige paremini, kuid kasutada võib ka teisi värve ja mustreid. Vesimärk võib olla kas ühe piksli suurune või tervet pilti kattev. Antud näiteks kodeerime me pildi vesimärgiga kindlas positsioonis, kasutades *Translate*-sõlme; see abistab meid pildi hilisemal dekodeerimisel, kuna me peame vesimärki otsima vaid kindlast kohast. Kasutame sõlme *RGB to BW* (RGB halltoonideks), et konverteerida pilt halltoonideks, ning sõlme *Map Value* (teisenda väärtus), et skaleerida vesimärgi heledusväärtused vahemikku 0.0-0.1. Sellega vähendame me vesimärgi intensiivsust kümnekordselt. Sõlme *Add* (liitmine) abil liidame vesimärgi kujutise kodeeritava pildile, muutes vastavad pikslid vaevumärgatavalt heledamaks.



Vesimärgi lisamine pildile kindlasse kohta

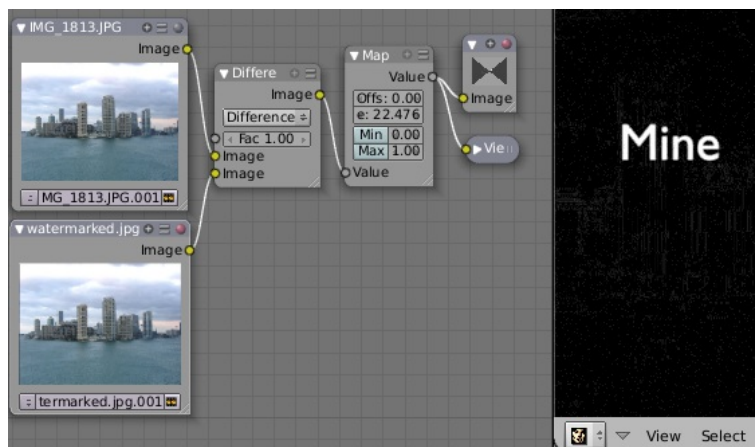
Kui sa soovid, et vesimärk oleks märgatav ka palja silmaga, siis võid ta teha ta kontrastsemaks või kasutada teisi toone. Vesimärgi võib lisada ka muudel viisidel, kasutades erinevaid segamissõlme funktsioone ning keerukamaid töötlusi. Proovi ja katseta!

Muud kasutusviisid

Seda tehnikat võid kasutada ka nähtavate efektide lisamiseks: näiteks pannes tiitrites sõnad ilmuma veepinnale või nõiamärgid kurjast vaimust vaevatud tüdruku käsivarrele.

Vesimärgistatud pildi dekodeerimine

Kui sa leiad pildi, mis võib olla sinu oma, kasuta allpool toodud kaarti, et võrrelda seda oma vesimärgistamata originaaliga. Vesimärgi kontrollimiseks määra *Mix* sõlme funktsiooniks erinevus (*Difference*) ning kasuta sõlme *Map Value*, et tulemust võimendada. Tulemus on suunatud kuvamissõlme ning pilti kodeeritud vesimärk on selgelt esile toodud:



Vesimärgi olemasolu kontrollimine pildil

Erinevad pildi pakkimise algoritmid lisavad pildile ebatäpsusi, mis tulemuses paistavad välja värvimürana. Katseta erinevate pakkimisseadete ja vesimärkidega, et leida kombinatsioon, mis töötab sinu jaoks kõige paremini. Katsetamiseks võid paigutada kodeeritava pidi ja sõlmekaardi ühte stseeni ning dekodeerimise sõlmekaardi teise stseeni. Muuda Blenderi salvestatava pildi seadeid ning ära unusta vesimärgistatud pilti pärast salvestamist uuesti laadida (vastasel juhul ei pruugi sa tehtud muudatusi näha). Toodud näites oli vesimärk selgelt märgatav vähemalt kuni JPEG-pakkimisalgoritmi kvaliteeditasemeni 50%.

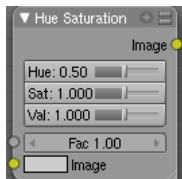
Dodge ja Burn (ala- ja ülesäritamine) segamifunktsioonide kasutamine

Kasuta *Dodge* ja *Burn* funktsioone koos maskiga, et mõjutada vaid pildi määratud osi. Vanadel headel pimikuaegadel kasutati pulga otsa kinnitatud papiõngast, et fotosuurendi all pildi teatud osade säritust vähendada. Vajalikele kohtadele varju heites langes sinna

vähem valgust ja need kohtad jäid tulemusena heledamad.

Vastupidise tulemuse saavutamiseks kasutati auguga pappi, et säritada vaid tumendamist vajavaid alasid. Maskis olevast august pääses valgus läbi ja "põletas" vastavas kohas pildi paberile. Samamoodi saame segamissõlme abil muuta pildi määratud alasid heledamaks või tumedamaks. Meeles tuleb pidada, et kuna tegu on pärandiga fotoprotsessist, siis rohkem valgust annab tulemuseks tumedama pildi ja vastupidi. Maski loomisel arvesta, et must värv tulemust ei mõjuta ning valge on maksimaalse mõjuga. Puuduvad veel ainult ilmuti lõhn, meeldiv muusika taustal ning voolava vee solin. Need olid vanad head ajad.

Toonimissõlm (*Hue Saturation*)



Alternatiivina RGB-värvisüsteemile võib värve kirjeldada ka tooni, küllastuse ja heleduse abil (*Hue, Saturation, Value*). Tooni võib vaadelda kui vahemikku 0.0-1.0 normaliseeritud värviringi. Lisatud värvi hulk sõltub lisatava värvi küllastusastmest. Mida kõrgem on küllastus, seda rohkem lisatakse tulemusele vastavat värvi. Kasuta küllastuse (*Saturation*) liugurit, et muuta pildi värve erksamaks.

See sõlm teisendab kõigepealt sisendpildi värvid RGB-süsteemist HSV-süsteemi ning muudab seejärel pildi tooni ja värvide erksust ehk küllastust:

Hue (toon)

Hue ehk tooni liugur kontrollib pildi värvitoonide nihet värviringil. Väärtus 0.5 (keskel) sisendpildi toone ei muuda. Liuguri nihutamisel vasakule muutuvad toonid sinakamaks ja edasi rohekamaks ning kollakamaks. Punane pilt muutub kõigepealt lillakaks, siis lillaks, siniseks ja lõpuks sinakasroheliseks. Liuguri nihutamine paremale muudab toonid alguses punakamaks ning seejärel rohekamaks. Sinine pilt muutub lillakaks, lillaks, punaseks, oranžiks ja seejärel kollaseks. Punane pilt muutub kuldseks, rohekaks ja seejärel sinakaks.

Sat: (küllastus)

Saturation ehk küllastus mõjutab värvide erksust. Väärtus 0 "eemaldab" pildilt kõik värvid ehk tulemuseks on halltoonides pilt. Väärtus 1.0 sisendpildi värvide erksust ei muuda ning 2.0 muudab värvid erksamaks. Tähele tasub panna, et küllastuse liigsel langetamisel või tõstmisel kaotame me osa värvinfost ning hiljem seda infot enam tagasi ei saa.

Val: (heledus)

Value ehk heledus mõjutab pildi üldist heledust. Suuremad väärtused kui 1.0 muudavad pildi heledamaks, väiksemad tumedamaks.

Fac: (kordaja)

Factor (kordaja) määrab, kui palju mõjutab sõlm sisendpilti. Kordaja 0 tähendab, et sisendit ei muudeta. Kordaja 1.0 on maksimaalne mõju ning 0.5 annab tulemuseks 50:50 segu originaalist ja maksimaalselt mõjutatud pildist.

Toonimissõlme nipid

Mõned asjad, mida selle sõlme kasutamisel silmas pidada:

Toonide töötlemine käib samaselt värviringiga.

Sinine sisendpilt ja tooni (*Hue*) seade 0 või 1 annab tulemuseks kollase ehk vastandvärv (tuleta meelde, et valge miinus sinine on kollane). Kollane pilt tooni seadega 0 või 1 annab tulemuseks sinise pildi.

Toon ja küllastus töötavad käsikäes.

Tooni väärtus 0.5 hoiab toonid samad, kuid küllastuse (*Saturation*) liugur muudab värve erksamaks või plassimaks.

Halltoonidel tooni kui sellise väärtus puudub ning neid tooni liuguri väärtus ei mõjuta.

Halltoonid on värvid, mille puhul RGB-kanalite väärtused on võrdsed. Halltoone saab mõjutada vaid heleduse (Val) liuguri abil. See kehtib kõigi halltoonide kohta vahemikus mustast valgeni.

Efekti muutmine ajas.

Tooni ja küllastuse väärtused on määratud küll liuguri abil, kuid mõjufaktorit saab muuta, kui ühendada tema sisendisse ajasõlm (*Time*).

Värvimine

Toonimissõlm nihutab vaid juba pildis olemas olevaid toone. Halltoonides pildi värvimiseks või värvi lisamiseks mõnele pildile kasuta segamissõlme (*Mix*) funktsioone.

Toonimise näide



Odava digifotoaparaadiga kehvades valgusoludes valguga tehtud pilt vajaks parandamist. Punakate toonide nihutamisel sinakamate poole parandame värve ning küllastuse vähendamisega muudame pildi silmale meeldivamaks. Heleduse liuguri abil muudame pildi veidi heledamaks.

Heleduse ja kontrasti sõlm (*Bright/Contrast*)



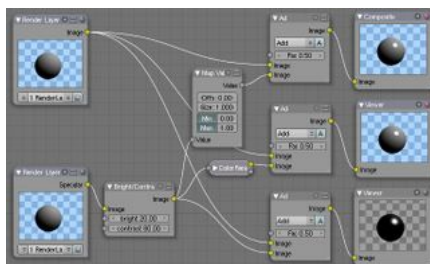
Lihtne näide

Bright (heledus)

Heleduse tõstmine liitmistehte abil, pildi väärtuste lineaarne tõstmine. Erineb toonimissõlme heleduse (*Value*) funktsionaalsusest, kuna ei kasuta alusena näivat heledust, vaid lihtsalt RGB-kanalite väärtusi. Negatiivne väärtus muudab pildi tumedamaks.

Contrast (kontrastsus)

Kontrasti tõstmine. Muudab heledad pikslid heledamaks ja tumedad tumedamaks. Põhimõtteliselt on tegemist lineaarse skaleerimisega halltoonide ümber. Kõrgem väärtus tõstab esile pildi detaile. Negatiivne väärtus vähendab pildi üldist kontrasti.

Märkused

On võimalik, et selle sõlme väljundis esineb piksliväärtuseid, mis on väljaspool normaalset 0.0-1.0 vahemikku. Kui sõlme väljundit on plaanis kasutada järgnevates operatsioonides, on mõistlik lõigata ebasobivad väärtused maha kas sõlmega *Map Value* (lülitades sisse valikud *Min* ja *Max*) või *ColorRamp* (vaikewäärtustel).

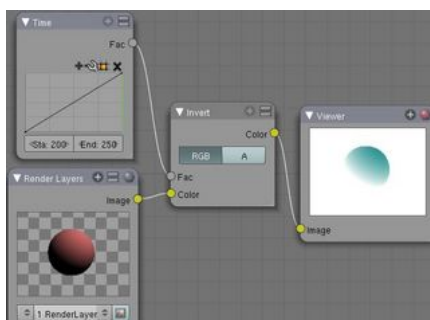
Mõlemad eelmainitud sõlmed skaleerivad pildi väärtused tagasi lubatud vahemikku. Juuresolevas näites (pildil) soovime me tõsta läike renderduskäigu intensiivsust. Alumine lõim näitab, mis juhtub, kui me ei eemalda vahemikust 0.0-1.0 väljapoole jäävaid piksliväärtuseid. Läikekihi tumedates osades on kontrasti tõstmise tõttu piksliväärtused negatiivsed ning kui need lisada hallile toonile, on tulemuseks must. Parandatud kontrastiga kihi korrigeerimine sõlmega *Map Value* või *ColorRamp* annab aga soovitud tulemuse.

Gamma

Gammakorrektsiooni eesmärgiks on parandada valgustusega seotud probleeme ning teisendada pilti lineaarsest logaritmiliseks ning vastupidi. Valgustuse probleemidest saab parandada valguse vaibumise probleeme, valguse/varju piirialasid ning heledate-tumede alade tasakaalu. Renderdamismootor kujutab endast virtuaalset kaamerat. Renderdatud pildile gammakorrektsiooni tegemisega saad sa parandada virtuaalsel pildistamisel tekkinud vigu. Sama protsess toimub ka digitaalsetes fotokaamerates. Gammakorrektsiooni väärtus on 0.45. Mitte 2.2.

Tekstuuri ja värvide gamma pöördkorrektsioonil on veel üks väga oluline tagajärg, kui sa kasutad stseenis kaudvalguse lähendamist (*Radiosity*, *Global Illumination*). Kaudvalguse arvutustes käsitletakse tekstuuride ja värvide RGB-väärtusi nende peegelduskoefitsientidena. Kui sa ei kasuta gamma pöördkorrektsiooni, on kaudvalgus liiga hele, sest pinnad peegeldavad stseeni tagasi palju rohkem valgust, kui nad peaksid.

Gammakorrektsioon toimub Blenderis kahes kohas. Esiteks värvisõlmedes - nii selles sõlmes kui ka *Tonemap*-sõlmes - ja teiseks kiirgavuse (*Radiosity*) arvutustes. Näitepildil on näha gammakorrektsiooni mõju värvidele.

Negatiiv (*Invert*)

See sõlm muudab pildi piksliväärtused vastupidiseks ning tulemuseks on negatiivpilt. Kasuta seda näiteks maskimise puhul

pöördmaski loomiseks.

Valikud

Factor (kordaja)

Seade, mis kontrollib sõlme mõju tugevust.

Color (värv)

Värv või sisendpilt. Antud juhul punane kera mustal läbipaistval taustal

RGB

Pöörab RGB-värvikanaleid (valge suhtes). Antud näites on punase vastandiks tsüaan (sinakasroheline).

A

Pööra ka läbipaistvuskanal (alfa ehk läbipaistvuse info). Käepärane maskide loomisel.

AlphaOver sõlm

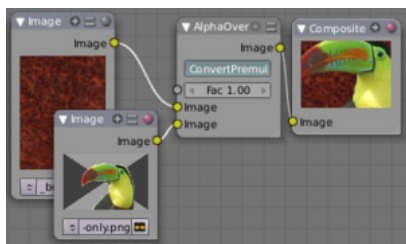


AlphaOver sõlm

Kasuta seda sõlme, et paigutada üks pilt teise peale. AlphaOver-sõlm võtab kaks sisendpilti ning kombineerib need alfakanali alusel väljundpildiks. Ühenda tagaplaan B ülemisse pessa ja esiplaan A alumisse. Aladel, kus esiplaani pikslite alfa on suurem kui 0 (teisistõnu on nad nähtavad), katab esiplaan tagaplaani (katvuse määr sõltub alfa väärtusest).

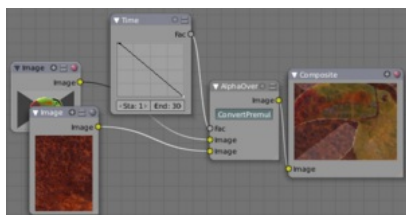
Kasuta liugurit Factor (kordaja), et määrata sõlme mõju tugevust. Tegemist on põhimõtteliselt alfakanali kordajaga: näiteks kui alfa on 0.8 ja faktor on 0.5, siis on tulemus sama kui alfa oleks 0.4. Väiksemad kordaja väärtused kui 1.0 muudavad esiplaani rohkem läbipaistvaks.

Näited



Komposiitpildi loomine *AlphaOver* sõlme abil

Selles näites on tuukani pilt paigutatud mustri tagaplaanile. Kasuta nuppu *PreMultiply* (eelkorutus), kui esiplaani pildi piksliväärtused pole veel alfaga korrigeeritud (*unpremultiplied*). Sa saad probleemist aru, kui esiplaani servadesse tekib kole helendav triip. Tulemuseks on komposiitpilt kahest sisendpildist.



Animeeritud läbipaistvuse efekt, kasutades *AlphaOver* sõlme - kaader 11

Selles näiteks kasutame me ajasisendi poolt mõjutatud kordaja väärtust läbipaistvuse animeerimiseks. Seda efekti võib kasutada näiteks läbi seinte vaatamise efekti loomiseks. Siin sulandame me *AlphaOver* sõlme abil tuukani pildi 30 kaadri jooksul sujuvalt taustaks oleva puidupinna peale. Pildil olevas kaadris hakkab tuukani pilt just nähtavale ilmuma. *AlphaOver* ei muuda sisendpiltide värve ning ei väljasta pilti, kui üks sisenditest on ühendamata.

Veidrad halod ja ümbrisjooned

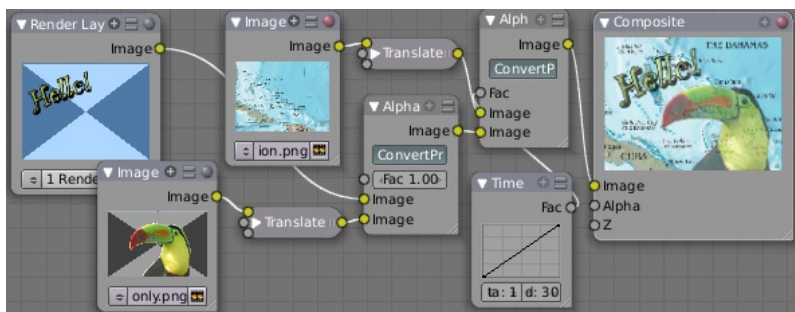
Veidi täpsustavat juttu nupu *PreMult* (eelkorutus) kasutamise kohta. Alfakanali (läbipaistvus) väärtused on vahemikus 0 kui 1. Kui sa muudad pildi poolläbipaistvaks, et seda teise pildi peale paigutada (vastasel juhul kataks esiplaan tagaplaani täielikult), korrutab *AlphaOver* sõlm pildi RGB-väärtused alfa väärtustega läbi. Näiteks kui alfa on 0, seatakse ka piksliväärtused nulliks.

Pildi A paigutamisel pildi B peale korrutatakse seega pildi A piksliväärtused läbi pildi A alfa väärtustega. Seejärel korrutatakse pildi B piksliväärtused läbi pildi A alfa täiendväärtusega (1-alfa), muutes pildi B läbipaistvaks nendes kohtades, kus A on läbipaistmatu ja vastupidi. Viimase sammuna liidetakse piltide A ja B piksliväärtused ja alfa väärtused kokku, et saada komposiitpilt.

Eelnevalt juba alfaga läbi korrutatud (*premultiplied*) piltide puhul ei tööta kirjeldatud protsess enam korralikult ning seepärast tulebki *AlphaOver* sõlmele selgeks teha, mis laadi sisendiga meil tegemist on ehk lülitada sisse nupp *Convert PreMult* (konverteeri

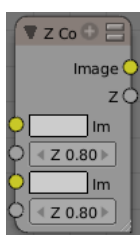
eelkorutus). Viga tuleb esile vaid poollõbipaistvates kohtades, enamasti servaaladel. Enamasti on probleemiks sõlmed, mis kombineerivad kahte pilti alfa alusel ning mille väljundit sa soovid kasutada järgnevates sõlmedes. Eelnevalt komposiititud pilt on juba alfaga läbi korrutatud ja eeldab seega nupu *Convert PreMult* (konverteeri eelkorutus) sisselülitamist.

Kui sa seda ei tee, on tulemuseks heledad halod esiplaani servaaladel. Alfaga läbi korrutatud pilt võib tulla ka renderdusmootorist, kui see on valikutes sisse lülitatud. Üldiselt on tänapäeval suurem osa alfa kanalit kasutavatest piltidest juba alfaga läbi korrutatud, kuid Blender miskipärast eeldab vastupidist. Läbikorrutamise eemaldamine (*unpremultiply*) on oluline, kui on vaja teha tugevat värvikorrektiooni. Vastasel korral töödeldakse servaalasid valesti.



Piltide komposiitimine kasutades *AlphaOver* sõlme

Sügavuse kombineerimise sõlm (Z-Combine)



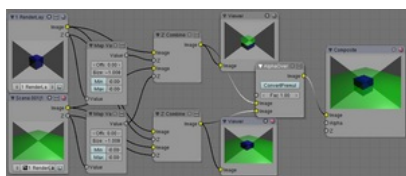
Sügavuse
kombineerimise
sõlm Z-
Combine

Sõlm *Z-Combine* kombineerib kaks pilti nende sügavusinfo (Z-kanali) alusel. Sügavusinfo määrab, millised esiplaani osad paiknevad ruumis eespool kui tagaplaani omad. Kui piksli sügavus on esi- ja tagaplaanil sama, kasutatakse esiplaani piksli. Väljundile lisatakse kombineeritud sügavusinfo kanal, mis võimaldab seda kasutada järgnevatel Z-kanalit kasutavates sõlmedes.

Sõlm *Z-Combine* valib esi- ja tagaplaani pikslitest lähima ehk selle, mille *Z*-väärtus on väiksem. Enamasti on objektid kaamera ees ning neil on positiivne *Z*-kanali väärtus. Kui ühe piksli *Z*-kanali väärtus on negatiivne ja teisel positiivne, siis kasutatakse esimest, sest selle väärtus on väiksem. Negatiivset *Z*-kanalit võib ette kujutada kui kaamera taga asuvat ala. Kahe negatiivse *Z*-kanali väärtuse puhul valitakse see, kumma väärtus on väiksem (ehk absoluutväärtus suurem).

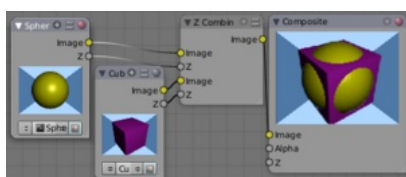
Alfakanali info kantakse sisendpildidest kombineeritult üle. Kasutatakse nii piksli pildilist kui alvakanali väärtust. Kui piksel on osaliselt või täielikult läbipaistev, siis paistab tagaplaan temast läbi ka hoolimata väiksemast Z-väärtusest. Teravate servade ja kontrastsete alade puhul silutakse alfakanalit automaatselt, et vältida vigaseid üleminekuid, kuid see võib anda poolläbipaistvate alade puhul ebaõige tulemuse.

Poolläbipaistvuse ja sügavusinfo õigeks kombineerimiseks võid kasutada kahte eraldi *Z-Combine* sõlme, millest üks on tavaline ja teine pööratud väärtustega. Pööramiseks võid kasutada näiteks sõlme *MapValue*, kus välja *Size* ehk sügavusinfo mõõtkava väärtuseks on määratud *-1.0*. Tulemused kombineeri kokku *AlphaOver* sõlmega.



AlfaOver ja Z-Combine sõlmede kasutamine.

Näited



Lähimate pikslite valik

Parempoolses näites on kaks renderdatud pilti kombineeritud sügavuskanali alusel *Z-Combine* sõlmega. Ühel pildil on kujutatud kera ja teisel kuupi. Kera ja kuup asuvad samas kohas ja kera on veidi suurem kui kuup. Kuubi lähim serv on lähemal kui kera ning seetõttu näeme me selles kohas kuupi. Kuna kera on veidi suurem kui kuup, siis ei mahu ta tervenisti kuubi sisse. Kohtades, kus kera lõikab kuubi külgi, on kera kaamerale lähemal kui kuup. Seetõttu on kera nendes kohtades sõlme *Z-Combine* väljundpildil näha.

Seda sõlme saab kasutada ka esiplaani ja tagaplaani keerukamaks kombineerimiseks. Walt Disney oli selles vallas teedrajav,

kasutades mitmele eraldi klaasile maalitud elemente, mille vahele paigutatud tegelased näisid paiknevat otse sündmustiku keskel. Kaamera liikumisega sai lisada ka muutusi perspektiivis, jättes mulje, et näiteks Bambi kepsleb läbi metsatuka ning kaamera liigub temaga kaasa.

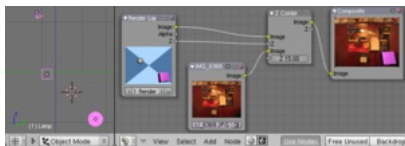
Lubatud sisendid

Z Input (Z-sisend) sisendipesad lubavad sisendina kõiki arvulisi väärtusi väljastavaid sõlmi. Pilti ootavad sisendipesad ja aktsepteeri sisendina värvust, sest sellel puuduvad UV-koordinaadid.



Sügavuskanali info kasutamine teisel pildil

Sügavuskanali infot võib kasutada ka teiste piltide kombineerimiseks, antud näites on sügavusinfo võetud kera ja kuubi piltidelt, kuid selle alusel kombineeritakse hoopis erinevaid mustreid.



Sõlme **Z-Combine** kasutamine määratud sügavustasandiga

Selles näites on renderdatud pilt kombineeritud fotoga ning fotole on käsitsi määratud sügavuse väärtus. Seda võib ette kujutada kui lamedat kaamera suunas orienteeritud objekti, näiteks paberilehte. Lilla kuup on kaamerast 10 ühiku kaugusel ja hall pall 20 ühiku kaugusel. 3D-kursor on kaamerast umbes 15 ühiku kaugusel. Määrame foto sügavuskauguseks 15 ühikut ehk paigutame ta kuubi ja palli vahele. Kombineeritud pildil paikneb kuup laual ja pall n-õ. laua all.

Läbipaistva mehe efekt

Kui sa valid esiplaani pildi, mille alfa väärtus on suurem kui tausta oma ning kasutad **Z-combine** sõlme koos kergelt suurendatud taustaga, siis läbipaistva ala piirjoon moonutab taustapilti. Ta teeb seda täpselt nii palju, et jääks mulje, nagu näeksime tausta läbi läbipaistva objekti.

Värvitasakaalu sõlm (**Color Balance**)

Värvitasakaalu (**Color Balance**) sõlm võimaldab muuta pildi värve ja heledust kahe erineva funktsiooni alusel.

Lift, Gamma, Gain funktsioon kasutab Lift (baastase), Gamma (gamma) ja Gain (kordaja) arvutust pildi korrigeerimiseks. Lift tõstab musta taset, Gamma mõjutab halltoone ja Gain heledaid toone.

The Offset, Power, Slope valem kasutab peaaegu samalaadset operatsiooni, kuid teiste nimede all. Tegemist on ASC (*American Society of Cinematographers*) poolt välja töötatud formaadiga (CDL ehk Color Decision List) värvikorrektuuri alase info vahetamiseks. Iga värvikanali kohta on kolm komponenti. Offset (nihe) määrab baastaseme, Power (aste) gammafunktsiooni ja Slope (tõus) kordaja. Arvutatakse see valemi järgi

$$\text{out} = (i * s + o)^p,$$

kus

out = väljundpiksli väärtus

i = sisendpiksli väärtus (0=must, 1=valge)

s = kordaja (ükskõik milline positiivne arv; vaikimisi 1.0)

o = baastase/nihe (ükskõik milline arv; vaikimisi 0)

p = kordaja (ükskõik milline arv, mis on suurem kui 0; vaikimisi 1.0)

Factor (kordaja)

Kordaja, mis määrab sõlme mõju tugevuse.

Toonikorrektuuri sõlm (**Hue Correct**)

Toonikorrektuuri (**Hue Correct**) sõlm võimaldab muuta pildi tooni, küllastuse ja heledust kõvera abil.

Vaikimisi on kõver sirge ehk väärtused kantakse üks-ühele üle. Sõlm võimaldab mõju igale värvitoonile eraldi määrata. Tooni, küllastuse või heleduse taseme muutmiseks nihuta kõvera punkte üles või alla. Pikslite väärtusi muudetakse vastavalt nende toonile ehk võimalik on langetada näiteks vaid kollaste toonide küllastust.

Tooniteisenduse sõlm (**Tone Map**)

Tooniteisendus (*tone mapping*) on tehnika, mille puhul paigutatakse suurema dünaamilise ulatusega info väiksema dünaamilise ulatusega pildile. Näiteks on keeruline samaaegselt pildile püüda detaile nii väga heledates päikeselaikudes kui ka tumedates varjualades. Mitme eraldi säritatud pildi kombineerimisel on võimalik luua suurema dünaamilise ulatusega pilt, kus kogu info on olemas. Selle info esitamine näiteks arvutiekraanil on aga siiski keeruline. Siin tulebki appi protsess "tone mapping" (tooniteisendus), mis surub suurema dünaamilise ulatusega info (näiteks vahemik 0.0-100.0) väiksemasse vahemikku (0.0-1.0). Selleks kasutatakse keerukaid algoritme, mis analüüsivad pildi heleduste jaotust ning loovad uue pildi mittelineaarse ülekande abil. Taoliseks tööks ongi mõeldud tooniteisenduse sõlm.

Tooniteisendus (*tone mapping*) üritab parandada probleeme, mis kaasnevad dünaamilise ulatuse lineaarse kokkusurumisega (näiteks **Map Value** sõlme abil) - kontrasti vähenemine, värvide kirkuse kadumine jne.

Tooniteisenduse sõlm kasutab kahte erinevat algoritmi:

Rh Simple

Key (võti)

Väärtus, millega samastatakse pildi keskmine heledus.

Offset (nihe)

Väärtus, millega saab kontrollida heleduse ülekandekõverat. Vaikeväärtus on 1.

Gamma

Gamma ehk astmefunktsioon, muudab halltoonide tasakaalu. Vaikimisi 1.

R/D Photoreceptor

Intensity (intensiivsus)

Kui selle väärtus on väiksem kui 0, siis muudab pildi tumedamaks, ja vastupidi.

Contrast (kontrastsus)

Kui väärtuseks määrata 0, hinnatakse kontrastsus sisendpildilt.

Adaptation (ühtlustamine)

Heleduste ühtlustamise ulatus. 0 on globaalne, 1 iga piksli puhul individuaalne.

Color Correction (värvikorreksioon)

Määrab, kas värvikanaleid parandatakse koos (väärtus 0) või eraldi (väärtus 1)

Komposiitimise vektorsõlmed (*Vector*)

Vektorsõlmed võimaldavad manipuleerimist vektorinfot sisaldavate kihtide, näiteks pinnanormaalidega. Need sõlmed on kasulikud ka siis, kui sa vektorarvutusest mõhkugi ei mäleta.

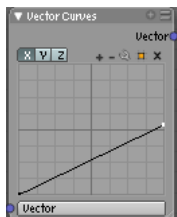
Vektorina käsitletakse Blenderis kahe või kolme väärtusega väljendatud parameetrit, näiteks RGB värvi (kolme komponendiga vektor), pinnanormaal (samuti kolme komponendiga, sest normaali suund on antud XYZ koordinaatides) ja piksli nihkevektor (kaks komponenti, sest ta väljendab nihet pildi tasapinna suhtes). Vektorid on olulised näiteks varjutuse arvutamisel ning objektide silumisel.

Normaalisõlm (*Normal*)

Normaalisõlm väljastab (pinna)normaali vektori ning selle skalaarkorrutise. Klõpsa sõlme aknas kujutatud keral, et määrata normaali suund.

Seda sõlme võib kasutada näiteks mõne vektorkihi väärtuste suunamiseks või värvisegamisoperatsioonides. Võtame näiteks *Color Mix* sõlme ja ühendame selle ühte sisendisse normaalisõlme ning teise mõne pildi. Vektori suunda muutes saame me muuta ka teise pildi värve. Põhimõtteliselt oleme me sedasi loonud omaenda värvipaleti, mis töötab samalaadselt mõnedes sõlmedes kasutatavale ringikujulisele värvipaletile.

Vektori ülekandekõverad (*Vector Curves*)

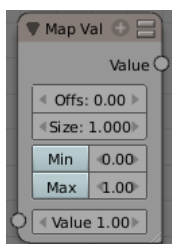


Ülekandekõvera sõlm võimaldab sisendvektorist genereerida väljundvektori graafiliselt määratud ülekandekõvera abil. Vektori kolm kanalit on kättesaadavad X, Y ja Z nuppude alt sõlme paneelil. Kõverale punkti lisamiseks klõpsa sellel sobivas kohas. Punkti liigutamiseks lohista seda hiirega.

Pane tähele, et punkti lohistamisel teisest punktist mööda vahetavad nad omavahel kohad.

Kasuta seda sõlme näiteks nihkevektori kihi muutmiseks, et mõjutada sügavusudu genereerimist.

Ülekandefunktsiooni sõlm (*Map Value*)



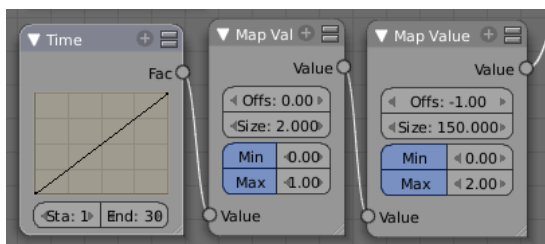
Ülekandefunktsiooni sõlm Map Value

Ülekandefunktsiooni kasutatakse väärtuste piiramiseks, skaleerimiseks ning nihutamiseks. See sõlm töötab järgnevalt:

- Offset (nihe) - see väärtus liidetakse sisendväärtusele
- Size (suurus) - sisendväärtus korrutatakse sellega
- Min/Max (miinimum/maksimum) - nendest väiksemad või suuremad väärtused lõigatakse väljundis maha määratud väärtusteni. *Min* ja *Max* tuleb eraldi sisse lülitada, klõpsates LMB vastavas linnukesekastis. Klõpsa ⇧ Shift LMB numbrikastil väärtuse muutmiseks.
 - Kui *Min* on sisse lülitatud, siis kõik väärtused, mis on väiksemad kui *Min*, saavad väärtuseks *Min*
 - Kui *Max* on sisse lülitatud, siis kõik väärtused, mis on suuremad kui *Max*, saavad väärtuseks *Max*

Seda sõlme on väga mugav kasutada näiteks sügavusudu loomiseks, kus sügavusinfo kanal (mille väärtused võivad olla näiteks vahemikus 0 kuni 500) tuleb skaleerida vahemikku 0-1, et see oleks sobilik udustamise sõlmede sisendiks.

Map Value sõlme kasutamine korrutamiseks



Ülekandefunktsiooni on mugav kasutada sisendväärtustest sulle vajaliku väljundväärtuse loomiseks. Juuresolevas näites annab ajasõlm 30 kaadri jooksul väljundiks väärtused vahemikus 0.00 kuni 1.00. Esimene ülekandefunktsiooni (*Map Value*) sõlm korrutab ajasõlme väljundväärtuse 2-ga, andes tulemuseks väärtuse muutuse vahemikus 0.00 kuni 2.00. Teine *Map Value* sõlm lahutab

väärtustest 1 (andes tulemuseks vahemiku -1.00 kuni 1.00) ning korrutab seejärel 150-ga, et saada üleminek -150 kuni 150 ja seda 30 kaadri jooksul. Veelgi lihtsam viis oleks kõigepealt sisendväärtused korrutada 300-ga ning lahutada sama sõlme abil 150. Pane tähele, et kõigepealt toimub korrutamise operatsioon ning seejärel liitmine.

Normaliseerimissõlm (*Normalize*)

Normaliseerib vektori nii, et vektori kogupikkuseks saab 1, kuid vektori suund säilib.

Komposiitimise filtrisõlmed (*Filter*)

Filtrid töötlevad pildi piksliväärtusi erinevate efektide loomiseks.

Filtrisõlm (*Filter*)



Filtrisõlm Filter

Filtrisõlm võimaldab mitmeid enamlevinud pildiparanduse ja töötamise operatsioone. Filtrite nimed, kui nad pole juba lennult mõistetavad, tulenevad enamasti nende väljamõtleja nimest:

Softener (pehmdus)

Muudab pildi kergelt uduseks, pehmeks.

Sharpen (teravdus)

Suurendab pildi kontrastsust, eriti servades.

Laplace

Muudab pehmemaks objektide servaalasid

Sobel

Loob negatiivse pildi, mis tõstab servad esile.

Prewitt

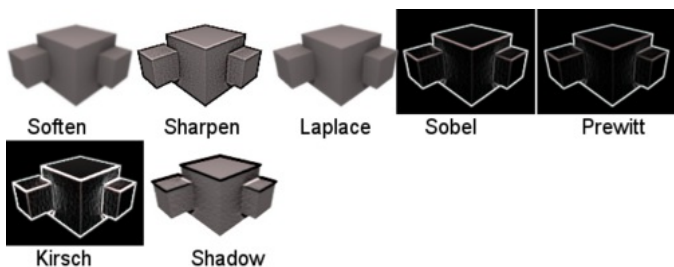
Teistsugune servaotsing.

Kirsch

Veel üks servaotsing.

Shadow (vari)

Varjude efekt, servade varjutus.



Filter sõlme erinevad funktsioonid.

Softener (pehmdus), Laplace, Sobel, Prewitt ja Kirsch tegelevad servaotsinguga erinevatel viisidel. Nende meetodite kirjeldustega võiks täita mitu suurt tahvli.

Udustamissõlm (*Blur*)



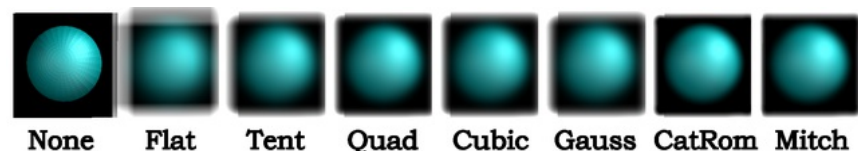
Udustamissõlm Blur

Udustamissõlm muudab pildi udusemaks. Valida on seitsme erineva udustamisviisi vahel. Udustamise raadiuse saab määrata X- ja Y-telje suunal eraldi. Vaikimisi on raadiuseks null ja udustamist ei toimu, seega udustamiseks peab väärtus olema nullist suurem. Võimalik on raadiuse väärtus sisestada ka teise pildi abil, selleks tuleb see ühendada suuruse (*Size*) sisendpessa. Selle sisendpildi piksliväärtused peaksid olema vahemikus 0 kuni 1, kuna nad korrutatakse omakorda veel X ja Y väärtustega.

Valikud

X ja Y väärtused määravad pikslite arvu, mille ulatuses udustamine toimub.

Nupp *Bokeh* (võib olla nähtav kui *Bok* või *Bo*) sunnib filtrit kasutama ringikujulist tööala (kernelit). See annab parema tulemuse, kuid on aeglasem kui vaikefilter, mis on ruudukujuline. Nupp *Gam* (ehk "gamma") laseb sõlmel teha enne udustamist pildi gammakorrektiooni.



Blur sõlm udu raadiusega 15% pildi mõõtmetest X ja Y suunas, ilma *Bokeh*/*Gamma* valikuteta. Klõpsa pildil, et erinevusi paremini näha

Erinevate udustamiste vahe on selles, kuidas nad mõjutavad servi ja gradiente ning säilitavad heledate ja tumedate alade vahetunde. Näiteks:

- Flat (tasane) udustab kõike ühtlaselt
- Tent (telk) säilitab heledad ja tumedad alad paremini
- Quadratic (ruutsõltuvus) ja *CatRom* säilitavad paremini teravaid servi
- Cubic (kuupsõltuvus) ja *Mitch* säilitavad heledad alad ning udustavad teravad servad

Suunatud udustamise sõlm (*Directional Blur*)

Udustab pilti vastavalt simuleeritud liikumissuunale. Seda kasutatakse enamasti lihtsamaks liikumishägu lisamiseks järeltöötluses.

Valikud

Iterations (iteratsioonid)

Kontrollib, kui palju kordi pilti efekti loomiseks kopeeritakse. Suuremad väärtused annavad sujuvama tulemuse.

Wrap (mähkimine)

Täidab udustamisest tekkivad tühjad alad.

Center (keskpunkt)

Määrab liikumise keskkoha - koha, mis seisab n-õ paigal. Keskkoha valik on oluline, kui kasutada liikumiseks pildi keeramist või suurenduse muutmist.

Distance (vahemaa)

Udustamise efekti tugevus.

Angle (nurk)

Udustamise suund

Spin (pöörlemine)

Pööramise efekt, pilti pööratakse määratud keskme ümber.

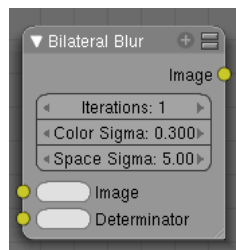
Zoom (suurendus)

Suurendamise efekt, mis suurendab pilti igal iteratsioonil.

Näide

Selles näites (.blend-fail on kättesaadav [siit](#)) on erinevad liikumishägu tüübid kenasti ära näidatud. Uuri seda ja muuda *Value* ja *Multiply* sõlmi, et muuta udustamise tugevust.

Bilateraalse udu sõlm (*Bilateral Blur*)



Udu sõlm Blur

See sõlm lisab pildile udu, kasutades selleks kõrgekvaliteedilist kohanduvat lähenemist. Seda võib kasutada näiteks:

- Blenderi kaudvarju kihi silumiseks
- n-õ erapooletute (*unbiased*) renderdajate (näiteks LuxRender või Maxwell) müraste piltide silumiseks
- mõnede efektide, näiteks uduste peegelduste ja pehmete varjude simuleerimiseks
- erinevate komposiitmisefektide loomiseks.

Sisendid

Sõlmel on kaks sisendit:

Image (pilt) pildi jaoks, mida udustada tahetakse.

Determinator (määraja): pilt, mis määrab udustamise piirid, ei pea olema ühendatud.

Kui ühendatud on vaid sisendpilt, siis määravad udustamise mõju sisendpildil olevad servad. Kui sisend *Determinator* on ühendatud, määratakse servad selle abil. Servade määramine eraldi pildi abil on kasulik, kui udustatav pilt on ise liialt mürane. Sel juhul saab puhta normaalse kihi ja sügavusinfo Z-kihi abil siiski servad täpselt ette anda.

Valikud

Iterations (iteratsioonid)

Kontrollib, kui palju kordi pilti efekti loomiseks kopeeritakse. Määrab põhimõtteliselt udustamise raadiuse.

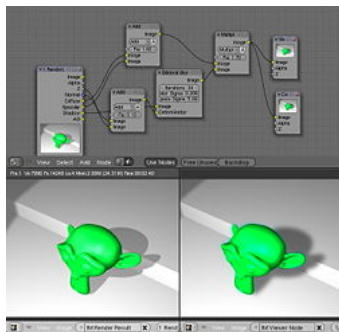
Color Sigma (värvierinevus)

Määrab läve, millest alates värvierinevust tuleks käsitleda servana.

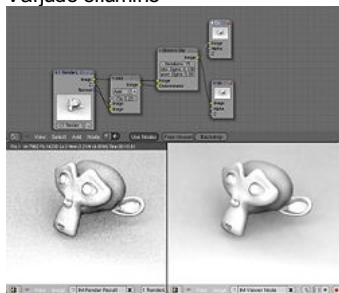
Space sigma (ruumierinevus)

Udustamise raadiuse täppiskontrolliks.

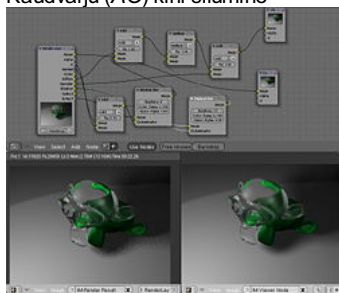
Näited



Varjude silumine

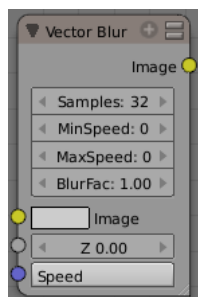


Kaudvarju (AO) kihi silumine



Udustatud peegeldused ja varjud

Vektorhägu sõlm (*Vector Blur*)



Vektorhägu sõlm

Vector Blur

Liikumishägu tekib, kui kaamera ees olevad objektid või kaamera ise filmimise ajal liiguvad. Udu tugevus sõltub liikumise kiirusest ja kaadri tekitamiseks kuluvast ajast (filmi puhul näiteks säriajast). Kuna arvutianimatsiooni ajal on ühes kaadris kujutatud vaid üksik ajahetk, mitte pisike ajavahemik nagu filmimisel või video salvestamisel, siis pole animatsioonis liikumishägu.

Liikumishägu lisamiseks on Blenderis kaks võimalust. Esimene võimalus (mis annab kõige parema tulemuse) on renderdada sama kaadri väikese ajanihkega kuni 16 korda uuesti ning seejärel need pildid üheks kokku liita. Seda võimalust saab sisse lülitada renderdamise seadetest. Tegemist on täpse, kuid väga aeganõudva protsessiga. Teine ja palju kiirem võimalus on kasutada *Vector Blur* sõlme.

Sõlme kasutamiseks ühenda tema sisenditesse sobivad renderduskäigud.

Märkus

Ära unusta renderdamisel lülitada sisse kiiruse (ehk *Vec*) renderduskäiku, millesse liikumise info salvestatakse.

MaxSpeed: maksimaalse kiiruse määranur. Võimaldab liikumishägu maksimaalset ulatust piirata. Liiga kiirest liikumisest tulenevate joonte ja muud pildimoonutuste põhjuseks on liiga kiirelt liikuvad pikslid. Et nende probleemide vastu võidelda, on sellel filtril minimaalse ja maksimaalse kiiruse sätteid, mille abil saab piirata seda, milliseid pikseleid udustatakse (nt kui piksel liigub väga-väga kiirelt, kuid sina oled maksimaalsele kiirusele määranud keskmise väärtuse, siis seda pikslit ei udustata).

MinSpeed: minimaalne udu ulatus. Võimaldab määrata, millisest kiirusest kiiremalt liikuvaid objekte udustatakse. Selle abil saab

näiteks jätta udustamata väga vähesel määral liikuvad, kuid siiski liikuvad objektid. Liikumiskiiruse ühikuks on pikslid. Väärtus 3 pikslit eraldab juba kenasti tausta ja esiplaani.

Näpunäide

Vektorhägu tulemuste sujuvamaks muutmiseks võid vektormaski ennast enne kasutamist veidi udustada, kuid pane tähele, et see võib anda ka kummalisi tulemusi. Kasuta seda vaid staatiliste piltide puhul, millel on palju liikumishägu.

Näited

Vektorhägu kasutamise kohta leiad pikema selgituse [siit](#).

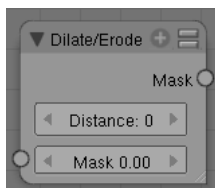
See sõlm kasutab liikumishägu loomisel [Blenderis uudset lähenemist](#). Kasuta vektorhägu sõlme liikumishägu loomiseks. See on ainus võimalus realistliku liikumishägu lisamiseks järeltöötluses. Selles [näidis .blend failis](#) on allapoole sirutuv taageldatud käsi, mis proovib palli üles korjata. Tulenevalt sellest, kuidas käsi liigub, on pilti tolles suunas udustatud. Kaamerale kõige lähemal olevaid sõrmi, mille Z-väärtus on kõige väiksem, udustatakse rohkem ning kaugemal olevat rannet udustatakse vähem.

Teadaolevad vead

Ei tööta korrektselt, kui pildiks on mitmekihiline OpenEXR

Blender 2.44+

Laiendamis-süvistamissõlm (*Dilate/Erode*)



Laiendamis-süvistamissõlm
Dilate/Erode

See sõlm võimaldab erinevaid kanaleid laiendada või sisse süüa. Operatsioon on kasulik näiteks maskide laiendamiseks või kitsendamiseks. Sisendpilt (milleks võib olla näiteks üks värvikanal või mustvalge pilt) ühendatakse sisendpessa Mask ja seade Distance (mõju ulatus) määratakse käsitsi. Positiivne Distance väärtus laiendab heledaid alasid. Negatiivne aga tumedaid.

Näide



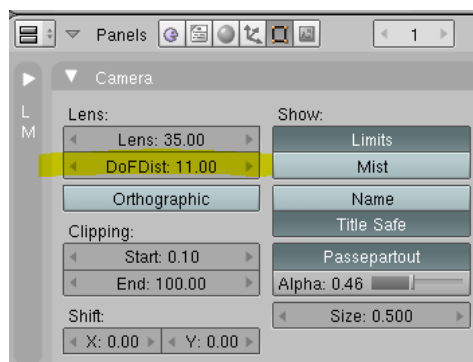
Lilla värvinüanss

Selles näites soovime me teha kuulikestega pildi väljanägemise huvitavamaks. Selleks laiendasime me punast kanalit ja ahendasime rohelist. Sinise jätsime muutmata. Nii punase kui ka rohelse laiendamisel oleks tulemuseks saanud... millise pildi (vihjeks: punane ja roheline annavad kokku kollase). Mõju tugevust saab muuta seade Distance (mõjuulatus) abil. [Näidisfaili saad alla laadida siit](#).

Sügavusudu sõlm (*Defocus*)

Selle sõlmega saad simuleerida udu, mis tekib "fookusest väljas" objektidel. Kuid seda võib kasutada ka muudel kunstilistel eesmärkidel. Sõlm udustab pilti teise pildi (enamasti sügavusinfo kanali Z) alusel.

Kaamera seaded (*Camera Settings*)



DoFDist seade

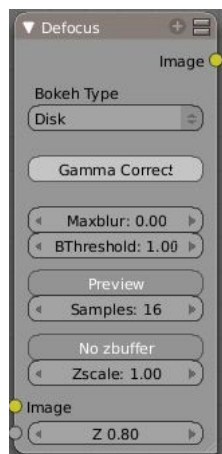
Defocus sõlm kasutab sügavusudu loomiseks stseeni kaamera seadeid, kui sisendpildiks on RenderLayer sõlm.

Fookuskauguse määramiseks on kaamera seadetes Depth Of Field (fookus, DoF) parameeter. Selle parameetri abil saab määrata fookustasandi kauguse kaamerast (objektid, mis on kaamerast DoFi kaugusel, on fookuses). Määra Depth of Field parameeter kaamera Object Data kaardil; sisestusvälja nimeks on seal Distance (vahemaa).

Et fookuspunkti stseenis nähtavaks muuta, lülita sisse valik Limits (piirid) kaamera esitusseadete (Display) alajaotuses. Teine

võimalus fookuskauguse määramiseks on määrata objekt, mis peab olema fookuses. Selleks klõpsa DoFi seadete juures oleva objekti valiku kastil ja määra sobiv objekt.

Sõlme sisendid



Sügavusudu sõlm
Defocus

Sõlm nõuab tööks kahte sisendit, udustatavat pilti ja sügavusinfo kanalit Z. Viimane ei pea olema reaalne sügavusinfot sisaldav pilt, vaid võib olla ka suvaline halltoonides kujutis või mask.

Sõlme seaded

Selle sõlme seadeteks on:

Bokeh Type rippmenüü

Siit saad sa määrata virtuaalse kaamera objektiivi iirise kuju ehk labade arvu. Iiris võib olla täiuslik ring (Circular) või näiteks kolme (Triangle), nelja (Square), viie (Pentagon), kuue (Hexagon), seitsme (Heptagon) või kaheksa labaga (Octagon). Rohkem valikuid pole lisatud seepärast, et üle kaheksa laba on juba eristamatu ringikujulisest (Circular) iirisest.

Rotate (pööra)

Võimaldab iirise kuju pöörata. Kasuta seda valikut, et iirise kujutist ümber oma telje keerata. Keeramise ühikuks on nurgakraadid.

Gamma Correct (gammakorrektsoon)

Töötab samamoodi kui Gamma valik udustussõlmes (Blur). Gamma seade on kasulik heledate alade intensiivsuse tõstmiseks, et luua realistlikumat muljet.



Defocus sõlm sügavusinfo
sisendiga

fStop (ava)

See on sügavusudu kontrollimisel kõige olulisem parameeter: ta määrab virtuaalse objektiivi avaarvu ja sellest tulenevalt ka sügavusteravuse ulatuse. Nagu päris kaameral, määrab ka siin *väiksem* number lahtisema ava ja seeläbi *väiksema* sügavusteravuse. Vaikeväärtus 128 on nii väike ava, et praktiliselt kõik on fookuses. Väärtuse poolitamine suurendab udu hulka poole võrra. Näiteks avaga 8 on sügavusudu kaks korda rohkem kui avaga 16. Avaarvu seade on välja lülitatud, kui sügavusinfo kasutamine (Use Z-buffer) on välja lülitatud.

Maxblur (maksimaalne udu)

Selle seade abil saad määrata maksimaalse udustamise määra. Ta määrab udu maksimaalse raadiuse pikslites. Maksimaalse raadiuse piiramine muudab protsessi veidi kiiremaks. Mida rohkem udu, seda aeglasem on selle loomine. Vaikeväärtus null tähendab, et maksimaalne raadius pole piiratud.

BThreshold

Sügavusudu sõlm (*Defocus*) pole täiuslik ning mõnikord võib udustamine lisada pildile ebameeldivaid efekte. Näiteks kui meil on fookuses olev objekt tugevalt udustatud taustal, siis võib tausta udu määrida ka fookuses oleva objekti servi. Halvim stsenaarium on fookuseeritud objekt väga kaugel tagaplaani puhul - kõrvuti olevate pikslite kaugused on väga erinevad ja tulemus kole. Sõlm püüab seda kompenseerida, kontrollides kõrvuti asetsevate pikslite kauguserinevust. Seade Threshold (lävi) määrab maksimaalse lubatud erinevuse, mille korral udu üle servade hajub. Või miskiit samast. Enamasti pole vajadust seda seadet muuta. Muuda seda vaid juhul, kui udustamisel tekivad probleemid, mida ehk saaks parandada.

Preview (eelvaade)

Nagu juba mainitud, võib udustamise protsess võtta omajagu aega. Et muuta seadete kruttimist interaktiivsemaks, on sõlmel eelvaate valik (Preview). Eelvaade renderdab tulemuse piiratud arvu pikslite pealt, mis on palju kiirem kui täielik töötus. Kuna töödeldakse vaid osasid piksleid, on tulemuseks mürane pilt. Müra hulka saab vähendada, suurendades töödeldavate pikslite arvu seade Samples (pildipunkte) abil. Eelvaade on vaikimisi sisse lülitatud. Kasuta eelvaadet, et sügavusudu seaded paika timmida, ning alles siis, kui kõik on valmis, lülita ta välja ja renderda pilt maksimaalse kvaliteediga.

Samples (punkte)

Eelvaate loomiseks loetavate punktide arv. Kasutatav vaid siis, kui eelvaade (Preview) on sisse lülitatud. Määrab eelvaate renderdamisel kasutatavate punktide arvu. Mida rohkem punkte, seda sujuvam tulemus, kuid seda rohkem võtab renderdamine aega. Eelvaate jaoks peaks vaikeväärtus 16 olema piisav.

Use Z-buffer (kasuta Z-puhvrit)

Mõnikord on vaja udustamise üle saada rohkem kontrolli, kui sügavusinfo kanal annaks. Näiteks siis, kui me soovime udustada vaid ühte valitud objekti, jättes teised puutumata, sõltumata nende kaugusest kaamerast. Seepärast võimaldab sõlm kasutada sügavusinfoks ka muid halltoonides pilte kui Z kanal. Näiteks võid sa ühendada sügavusinfo sisendpessa mõne pildi või maski, mis määrab udustamist vajavad alad. Või kui on vaja pildi udu hulka animeerida, võib sügavuse pessa ühendada ajasõlmega animeeritud halltoonpildi. Mõnikord saab parema sügavusudu tulemuse, kui päris sügavuspuhvri asemel kasutada ise loodud lihtsamat sügavuspilti. (Üks variant selle loomiseks on kasutada lihtsat gradienti, mis muudab pildi ühe serva teravaks ja teise uduseks. See lähenemine töötab päris hästi näiteks maastike puhul.) Üks ise loodud sügavuspildi eelis on ka sakitõrje võimalus, mida Z-puhvri ei ole. Lülita Use Z-buffer (kasuta Z-puhvrit) sisse siis, kui sügavusinfo sisendiks on renderdatud sügavuspuhver ning välja sisendiks on mõni muu pilt.

Zscale (Z-skaala)

Nähtav ainult siis, kui valik Use Z-buffer on välja lülitatud. Kui valik Use Z-buffer on välja lülitatud, määrab seade *Zscale* udustamise raadiuse. Enamasti on piltide väärtusvahemik 0.0 kuni 1.0, mis on udustamise raadiuseks liiga väike. Parameetri *Zscale* abil saab määrata, kui suur on udustamise raadius piksliväärtuse 1.0 ehk valge puhul. Seega, kui seade Use Z-buffer (kasuta Z puhvrit) on välja lülitatud, määrab see parameeter udustamise efekti tugevuse.

Näited



Selles [blend-näidisfailis](#) on kaamera vaateväljas rida kuule, millele on lisatud sügavusudu. Kaamera ava (*f-stop*) on 2.8, mis on päris väikese sügavusteravusega. Fookuskauguseks on määratud 7.5 ühikut. Mida kaugemal on kuulid, seda udusemad nad on.

Näpunäited

Eelvaade (Preview)

kasuta eelvaadet, et seaded paika timmida, ning lülita eelvaade välja vaid lõpprenderduse tarbeks. Sügavusudu renderdamine on arvutusmahukas.

Servavead

sügavusudu vigade vähendamiseks katsu stseen organiseerida nii, et üksteise ees asetsevad objektid poleks ruumiliselt teineteisest väga kaugel. Teine variant on renderdada esiplaani objektid eraldi renderduskihile ning kombineerida udustatud elemendid hiljem taustaga kokku.

"Animeeritud fookus"

pea meeles, et tegemist on järeltöötuse efektiga, mitte reaalse kaameraga. Mõned asjad, mida päriselus on lihtne saavutada, on järeltöötuses keerukad, kui mitte võimatud. Tüüpiline näide on stseen, kus osad objektid on kaamerale väga lähedal ja fookus on kusagil kaugemal. Päris elus muutuvad väikese sügavusteravuse kasutamisel väga ebateravad objektid servaaladel praktiliselt läbipaistvaks, lubades näha seda, mis jääb nende taha. Hollywoodi filmides kasutatakse seda efekti päris sageli. Kahjuks ei ole sama kvaliteetset tulemust järeltöötuses lihtne saavutada. Kui sa seda siiski teha soovid, siis siin on paar näpunäidet:

- Jaga oma stseen lähedal ja kaugel asuvateks objektideks ning renderda nad kahe eraldi käiguna.
- Kombineeri renderdatud elemendid kokku, lisades kummalegi oma eraldiseisva sügavusudu sõlme. Animeeri nüüd fookuspunkti kas kaamera seadete või sõlme seadete abil näiteks ajasisendiga. Sea sõlmekaart nii, et kui esiplaani udusus suureneb, siis tausta oma väheneb ja vastupidi. Selleks võid kasutada näiteks Map Value sõlme abi (kordajaga -1 saad pöördväärtused).

Sakilisus suure ava (väike *f-stop*) puhul

väikeste avaarvu seadete korral (vähem kui 5) võib täpselt fookuses olev objekt muutuda väga teravaks. Kui ta on kontrastse tagaplaani taustal, siis võivad esiplaani objektide servad muutuda sakiliseks. Selle probleemi vastu võitlemiseks:

- renderda kaks korda suurema resolutsiooniga ning vähenda pilti pärast kaks korda või
- udusta pilti väikese raadiusega (näiteks 2 pikslit) või
- muuda veidi fookuskaugust, nii et objekt ei ole päris täpselt fookuses.

- Kasuta suuremat avaarvu ja udusta pilti hiljem veel rohkem, kasutades *Blur* sõlme, kus udustamise raadiuse sisendiks on skaleeritud sügavusinfo kanal.
- Vaheta tagaplaan vähem kontrastse vastu.

Use Z-buffer (kasuta Z-puhvrit)


kuna sõlm ei suuda ise kindlaks teha, kas ühendatud on tegelik Z-kanal või mõni muu pilt, siis mõtle alati järele, kas valik Use Z-buffer on õiges asendis. Kui Zscale väärtus on juhtumisi suur ja sa lülitad sügavusinfo kanali kasutamise välja, võib tulemuseks olla plahvatuslikult kasvav renderdamise aeg. Ole sellega ettevaatlik.

Komposiitimise konvertersõlmed (*Converter*)


Nagu nimigi ütleb, tegelevad need sõlmed erinevat tüüpi konverteerimisoperatsioonidega. Nad võimaldavad pildikanaleid eraldada ning taas kombineerida ja seeläbi töötada iga kanaliga eraldi. Toetatud on mitmed erinevad värvikanalite grupid, nii traditsioonilised RGB ja HSV kui ka uuemad HDMI jms videostandardidest tulenevad kanalite jaotused.

Värviteisenduse sõlm (*ColorRamp*)

ColorRamp sõlme saab kasutada väärtusvahemike toonimiseks gradiendi alusel. Tema tööpõhimõte on samalaadne [värvüleminekute kasutamisele tekstuurides ja materjalides](#). Sisendisse (*Fac*) ühendatud pildi heledusväärtuste alusel määratakse gradiendi abil väljundpildi piksliväärtused ja alfakanal.

Vaikimisi on *ColorRamp* sõlme gradiendil määratud kaks positsiooni kummaski otsas: must ja valge. Must on vasakul ja valge paremal ning kummagi läbipaistvuse väärtus (alfa) on 1.0. Värvipositsiooni valimiseks ja muutmiseks klõpsa LMB  gradiendil olevale kitsal triibul. Näidispildil on must värv valitud ja valgelt esiletõstetud. Valitud värvipositsiooni jaoks kuvatakse sõlme paneelil tema värvikaart, alfa väärtus ja interpoleerimise funktsioon.

Värvipositsiooni tooni ja heleduse muutmiseks klõpsa LMB  aktiivse positsioonitähise värvikaardil ning määra avaneval paletil sobiv värv. Valiku kinnitamiseks vajuta ↵ Enter-

Uue positsiooni lisamiseks hoi a all Ctrl ja klõpsa Ctrl LMB  gradiendil soovitud kohas. Määra positsiooni värviseaded, klõpsates tema värvikaardil. Läbipaistvuse (*Alpha*) muutmiseks on avaneval värvipaletil omaette liugur tähisega A. *ColorRamp*-sõlme abil loodud läbipaistvuskanaali saad kasutada ka maskide loomiseks, kui ühendad vastava väljundi (*Alpha*) näiteks segamissõlme (*Mix*) sisendisse.

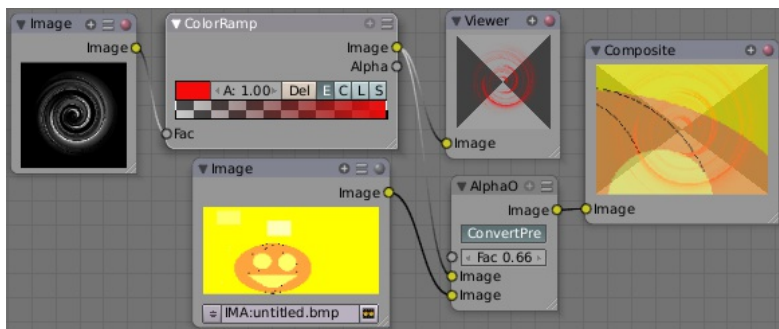
Värvipositsiooni eemaldamiseks tuleb see valida ning seejärel vajutada nuppu *Delete*.

Värvide üksteiseks üleminekut saad kontrollida interpoleerimisfunktsiooni abil. Võimalikud valikud on Ease, Cardinal, Linear ja B-Spline ja Constant. Erinevad funktsioonid annavad üleminekule erineva kiiruse ja ulatuse.

Kasuta nuppu A; et määrata iga värvi puhul selle läbipaistvust.

ColorRamp sõlme kasutamine maski loomiseks

Üks võimalus *ColorRamp* sõlme kasutamiseks on maskide loomine kahe pildi kombineerimise jaoks. Juuresolev sõlmediagramm näitab, kuidas *ColorRamp* sõlme maski loomiseks kasutada:



ColorRamp sõlme kasutamine maski loomiseks

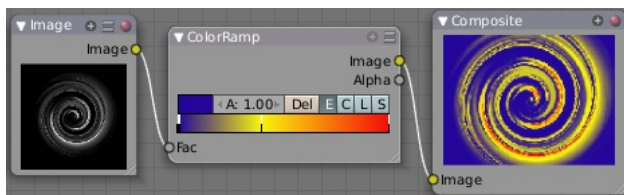
Siin kasutatakse mustvalget kujutist, millel puudub alfakanal, kahe pildi kombineerimisel maski loomiseks. Selleks on kasutatav kujutis ühendatud *ColorRamp* sõlme *Fac*-ühendusessa. (Kui me kasutaksime sisendina värvilist kujutist, siis oleks soovitatav see enne konverteerida mustvalgeks *RGB-to-BW* sõlme abil.)

Määrame *ColorRamp* sõlme vasakpoolse värvipositsiooni läbipaistvuseks 0.0 ja parempoolse värviks punase. Nagu näha, väljastab *ColorRamp* sõlm maski, mis on täiesti läbipaistev nendes kohtades, kus sisendpildi pikslid on musta värvi. Must on väärtus 0.0 ning *ColorRamp* sõlm määrab selle vasteks gradiendi vasakpoolse otsa värvi, mille me seadsime läbipaistvaks. Aladel, kus sisendpilt on valge (1.00), on vasteks gradiendil punane värv ehk parempoolne ots.

Et kontrollida väljundpildi läbipaistvuskanaali olemasolu, kombineerime ta kena kõrvitsa pildi peale.

ColorRamp sõlme kasutamine pildi värvimiseks

ColorRamp sõlme tugevus peitub selles, et me saame talle lisada mitmeid värvipositsioone. Järgnevas näites muudetakse igav mustvalge pilt leegitsevaks pööriseks!

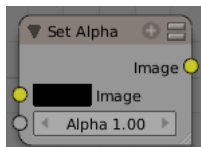


Siin on erinevatele halltoonidele määratud vastama kolm värvi: sinine, kollane ja punane. Kõigi katvuseks on määratud läbipaistmatu (katvus 1.00). Aladel, kus sisendpilt on must, on gradiendil määratud sinine värv. Halltoonide puhul valitakse gradiendilt vastavalt kas sinakas, kollakas või punakas toon. Täiesti valgete pikslite puhul on vastavaks värviks gradiendil punane.

RGB mustvalgeks muutmise sõlm (*RGB to BW*)

See sõlm konverteerib värvilise RGB-sisendpildi mustvalgeks.

Läbipaistvuse määramise sõlm (*Set Alpha*)



Läbipaistvuse määramise sõlm *Set Alpha*

See sõlm lisab pildile alfa ehk läbipaistvuse kanali. Mõned pildiformaadid, nagu näiteks JPEG, ei toeta alfakanalit. Et kombineerida JPEG pilti taustaga, peab talle lisama alfakanali.

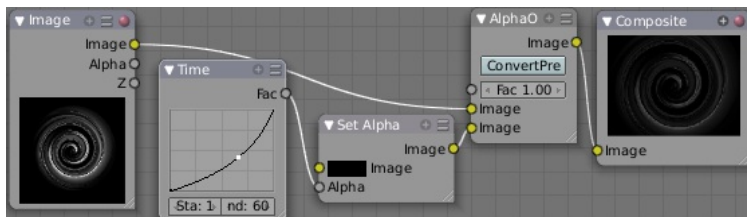
Sisendpesa Image (pilt) kasutamine on valikuline. Kui sisendpilti pole sõlmega ühendatud, saab väljundpildi värvi määrata värvikaardi abil. Värvu muutmiseks klõpsa LMB värvikaardil ja vali värvipaletilt sobiv värv.

Alpha (läbipaistvuse) väärtuse saab numbriliselt määrata kogu pildile ühiselt (1.00 läbipaistmatu ja 0.00 täielikult läbipaistev). Lisavõimalusena saab alfakanali määrata mõne teise pildi abil. Selleks tuleb sobiv pilt ühendada *Alpha* ühenduspessa.

Pane tähele, et see sõlm ei ole otseselt mõeldud enamlevinud komposiitmisoperatsioonide jaoks. Kui võimalik, siis kasuta alfakanali loomiseks sobivamaid meetodeid, näiteks maskimine värvi (*Chroma Key*) või tausta erinevuse (*Difference Key*) alusel. Seda sõlme on mõtet kasutada, kui mingil põhjusel pole võimalik teiste meetoditega korralikku alfakanalit luua.

SetAlpha sõlme kasutamine pildi mustaks sulatamisel

Videomontaažis kasutatakse ühest stseenist või plaanist teise üleminekul sageli mustaks sulatamist (*fade to black*). Nagu nimigi ütleb, muutub sel puhul pilt sujuvalt mustaks või vastupidi, mustast tagasi. Sulatada võib ka näiteks valgeks või mõneks muuks värviks, kuid must on hea neutraalne värv ning võimaldab silmadele puhkust anda ja vaataja meeled uuteks elamusteks valmis panna. Juuresolev sõlmeskeem näitab, kuidas kasutada *SetAlpha* sõlme mustaks sulatamise efekti loomisel.



Mustaks sulatamine

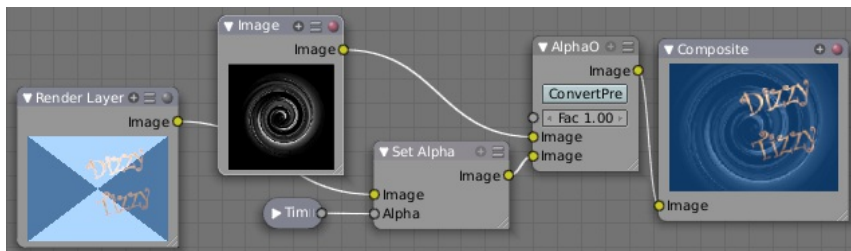
Antud juhul ignoreeritakse pöörise pildi alfakanalit. Selle asemel loome [ajasõlme](#) ja *SetAlpha* sõlme abil sujuvalt läbipaistvast läbipaistmatuks muutuva pildi. Ajasõlme kõver on eksponentsiaalse funktsiooni kujuga, seega muutub läbipaistvus alguses aeglasemalt ning lõpuolele kiireneb. *SetAlpha* sõlm ei vaja värvi määramiseks eraldi sisendpilti, selle asemel määrame värvikaardi abil värviks musta. *SetAlpha* sõlm kasutab sisendfaktorit ja värvi, et luua must aeglaselt läbipaistvast läbipaistmatuks muutuv kujutis. Läbipaistvuse muutus kestab 60 kaadrit. Mustaks sulatav pilt ja meie loodud animeeritud läbipaistvusega pilt on kombineeritud *AlphaOver* sõlme abil, et luua kombineeritud pilt. *SetAlpha* sõlm loob vastavalt renderdatavale kaadriks musta pildi, mille läbipaistvus on määratud ajakõvera abil. Tulemuseks on animatsioon kahe sekundi jooksul mustaks sulatatud pildist.

Stseeniinfot siin näites ei kasutatud

See näidis ei kasutanud oma töös sisendina renderduskihte. Selle kahesekundilise animatsiooni loomiseks ei kasutatud stseeniinfot, see loodi puhtalt Blenderi komposiitori vahenditega. See on üks näide, kuidas kasutada Blenderi komposiitorit eraldi modelleerimise ja animatsiooni võimalustest. (Mustaks sulatatava sisendpildi asemel võib sisendiks kasutada ka renderduskihte.)

SetAlpha sõlme kasutamine tiitrite pildile sulatamiseks

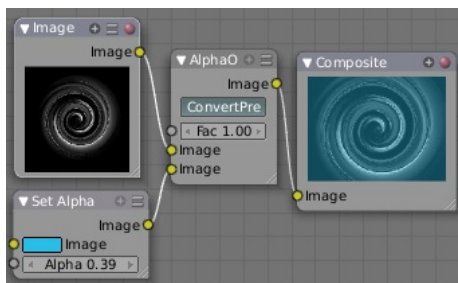
Sageli on vaja lisada pildile tiitreid või subtiitreid. Tiitrid võivad kaadrisse siseneda nii animatsiooniga (näiteks alt üles) kui ka sisse sulades. Tiitrite sisse sulatamiseks kasuta *SetAlpha* sõlme koostöös ajasõlmega (*Time*).



SetAlpha kasutamine tiitri sissesulutamisel

Selles näites määrab ajasõlme kõver alfa väärtuse. Renderduskiht tiitri tekstiga on sisendpildiks, mida me tausta peale sulatame. *AlphaOver* sõlmega segame me kaks pilti alfakanali alusel kokku. Pane tähele, et kuna Blender kasutab oma töös ja piltide genereerimisel läbikorrutatud piksliväärtuseid, siis peab nupp *ConvertPre* (konverteeri eelkorutus) nupp olema välja lülitatud. Selle peaksime me sisse lülitama, kui kasutame sisendiks teadaolevalt alfaga läbi korrutatud pilti (võib pärineda näiteks mõnest failist).

SetAlpha sõlme kasutamine pildi toonimiseks

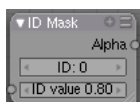


SetAlpha sõlme kasutamine pildi toonimiseks

Juuresolevas näites kasutatakse sinist pilti mustvalge pildi toonimiseks. Sinine pilt on loodud *SetAlpha* sõlme abil.

Toonimise tugevust saad juhtida numbrivälja Alpha väärtuse abil *SetAlpha* sõlmes. Ühenda *SetAlpha* sõlme väljund *AlphaOver* sõlme, et kombineerida toonitav pilt ja genereeritud tooniv pilt. Pane tähele, et nupp *ConvertPre* (konverteeri eelkorutus) on antud juhul sisse lülitatud.

ID Mask sõlm

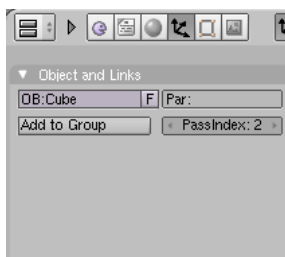


ID Mask sõlm

See sõlm kasutab objekti indeksite infot sisaldavat renderduskäiku, et luua sakitõrjega alfa mask valitud indeksiga objektidele. Mask on läbipaistmatu nende kohtade peal, kus on näha objekti, ja vastupidi. Kui objekt on osaliselt poolläbipaistev, siis on ka alfa mask vastavalt poolläbipaistev. See sõlm teeb sakitõrjet objekti indeksi kanalile ja silub objektide servaalade piksleid.

Objekti indeks

Objektiindeksid on kättesaadavad vaid vastava renderduskihi kaudu või siis näiteks OpenEXR-i failist (juhul kui vastav kiht on sinna salvestatud).

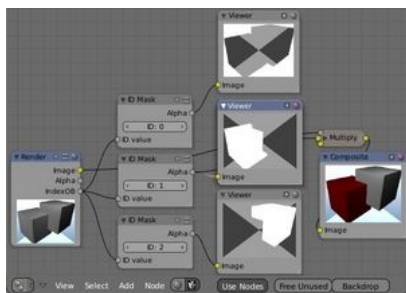


Objekti indeksi määramine

Igale objektile saab Blenderis määrata oma indeksi, juuresoleval joonisel on näidatud, kuidas seda teha. Kui renderdamisel on objekti indeksite kiht sisse lülitatud, on *ID Mask* sõlme abil võimalik meile vajaliku indeksiga objektide maskid eraldada.

See sõlm sobib näiteks sakkide probleemi eemaldamiseks sügavusudu genereerivatest sõlmedest.

Näide

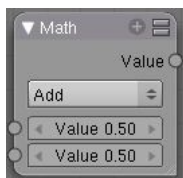


Näide

Selles näites on vasakpoolne punane kuup indeksiga 1 ning parempoolne kuup indeksiga 2. Kohas, kus kuubid lõikuvad, on näha kuupide ülemineku sakilisus. Kasutades silutud servadega maski objektilt indeksiga 1, saame me segamissõlme (*Mix*) abil korrutustehetega (*Multiply*) kätte meid huvitava objekti koos ilusate ühtlaste servadega. Seda objekti võime nüüd kasutada oma täiuslikus animatsioonis. Kalliskivi peal teadupärast sammal ei kasva.

Pane tähele, et mask on valge nendel aladel, kus objekt on täielikult nähtav ja läbipaistmatu, ning must seal, kus objekt on täielikult läbipaistev või mõne muu objekti poolt varjatud. Kui mõni muu objekt asub meid huvitava objekti ees, siis on mask selles kohas must. Seda isegi siis, kui varjav objekt on osaliselt läbipaistev.

Matemaatika sõlm (Math)



Matemaatika sõlm
Math

See sõlm võimaldab sisendpildi pikslitega sooritada erinevaid matemaatilisi tehteid. Toetatud on kõik enamlevinud matemaatilised funktsioonid. Kui sõmele on määratud vaid üks sisendpilt, siis liidetakse funktsiooniga genereeritud väärtused sisendpildi piksliväärtustele. Vali sobiv funktsioon rippmenüüst. Vaikimisi on valitud funktsioon "Add" ehk liitmine.

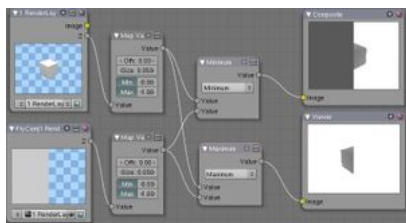
Funktsioonid *Sine* (Siinus), *Cosine* (Koosinus) ja *Tangent* (Tangens) kasutavad sisendina vaid ülemist ühenduspesa ning võimalikud sisendväärtused on vahemikus 0 kuni 2π - ehk radiaanid täisringi ulatuses.

Teadaolev viga: kui ühendatud on vaid üks sisendpilt, siis peab ta olema ülemises ühenduspesas.

Blender 2.44+

Näited

Z-maski loomine käsitsi

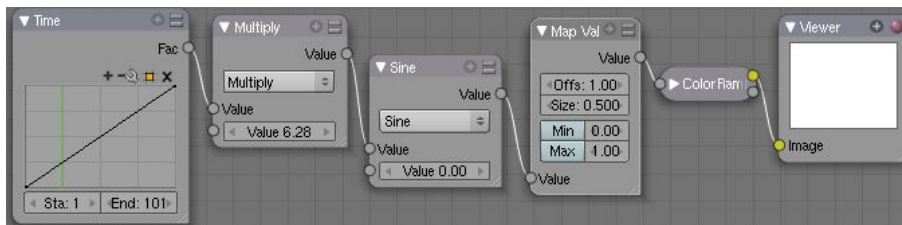


Näide

Selles näiteks kasutame kahte pilti. Esimesel on kuup, mis asub kaamerast umbes 10 Blenderi ühiku kaugusel. Teisel pildil on tasapind, mis katab pildi poolest ulatuses ja on kaamerast 7 Blenderi ühiku kaugusel. Mõlema pildi sügavuskanal Z on söödud läbi *Map Value* sõlme, et suruda nende väärtuste skaala 20-kordselt kokku (väli *Size*) vahemikku 0.0-1.0. Ühtlasi eemaldab *Map Value* sõlm väljapoole lubatud vahemikku jäävad väärtused (need asendatakse kas 0.0 või 1.0-ga).

Järgnev matemaatikasõlm (*Math*) miinimumfunktsiooniga (*Minimum*) valib kummastki sisendpildist väiksema piksliväärtuse ehk kaamerale lähema. Pane tähele, et sama tulemuse saaks ka segamissõlmega (*Mix*), valides seal funktsiooniks tumendamise (*Darken*). Tagaplaani sügavuse väärtus on lõpmatuses (praktiliselt on kaugus piiratud renderduspuhvri või salvestatud pildi bitisügavusega), seega lõigatakse ta 1.0-ks ehk valgeks. Teise *Math* sõlme abil leiame maksimaalsed sügavuse väärtused ehk saame kätte tausta ja kuubi selle osa, mis on tasandi taga.

Siinusfunktsiooni kasutamine pulseeriva pildi loomiseks

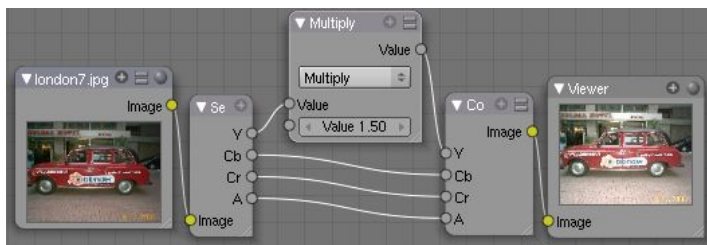


Selles näites kasutame matemaatikasõlme (*Math*) sõlme sisendina ajasõlme (*Time*), mis väljastab ajavahemikus 1-101 kaadrit lineaarselt väärtused 0 kuni 1. Roheline joon ajasõlme graafikul näitab hetkekaadrit. Praegu on selleks 25 ja ajasõlm väljastab hetkel väärtuse 0.25. See väärtus korrutatakse 2π -ga ja konverteeritakse omakorda siinusfunktsiooni poolt vahemikku -1 kuni 1.

Et saada genereeritavad väärtused lubatud vahemikku 0.0-1.0, selleks saadame väljundi läbi teisendussõlme (*Map Value*). Kõigepeal liidame me väärtustele 1 (saame vahemiku 0 kuni 2) ja seejärel korrutame 0.5-ga ehk saamegi vahemiku 0.0 kuni 1.0. Kasutame *ColorRamp* sõlme, et konverteerida need väärtused halltoonides pildiks. Tulemusena saame, et siinusfunktsiooni poolt väljastatud 0.0 on lõpp-pildil keskmine halltoon, must on -1.0 ja valge 1.0. Nagu näha, on $\sin(\pi/2)=1.0$. Võid seda kasutada nagu oma isiklikku värvikalkulaatorit! Animeerituna väljastab see sõlmekombinatsioon sujuva tsüklilise animatsiooni, kus pilt sulab mustast valgeks ja vastupidi.

Võid seda kasutada näiteks monteerimisel sulatamise efekti loomiseks või mõne muu maskina. Üks võimalus on kasutada seda pilti sisendina fookusesõlmele (*Defocus*), et muuta pilti uduseks ja tagasi. Muutes *ColorRamp* sõlme seadeid, võid teha ka pulseerivaid värve.

Pildikanali skaleerimine



Siin näites on kasutatud korrutamistehet (*Multiply*) pildi Y-kanalil ehk video heleduse komponendil, et muuta pilti heledamaks. Pane tähele, et sa pead kasutama *Map Value* sõlme, kus *Min* ja *Max* piirangud on sisse lülitatud, väärtuste lubatud vahemikku skaleerimiseks. Sel viisil võid logaritmfunksiooni kasutades skaleerida lubatud vahemikku ka suurema dünaamilise ulatusega, nn HDR (*High Dynamic Range*) pilt. Antud näite puhul võiks sama tulemuse saavutamiseks kasutada ka heleduse ja kontrasti (*Brighten/Contrast*) sõlme, mida on veidi lihtsam käsitseda.

Kvantiseerimine ja värvipaleti piiramine

Selles näites tahame me piirata pildi värvide hulga 256 võimalikuga. Võime seda kasutada näiteks selleks, et näha, milline näeks meie pilt välja 256 värviga piiratud mobiiltelefoni ekraanil. Et piiratud värvide hulgaga tulemust saavutada, peame me iga värvikanali lubatud väärtuste arvu piirama nii, et neid kombineerides oleks tulemuseks mitte rohkem kui 256 kombinatsiooni ehk värvi. Võimalike värvide arvu saame kõigi kanalite lubatud väärtuste arvu korrutades: $Q = R * G * B$.

Kuna meil on kolm värvikanalit ja 256 värvi, siis on meil võimalik valida erinevate kombinatsioonide vahel, kuidas $R*G*B$ saaks kokku 256. Näiteks kui $\{R, G, B\} = \{4, 4, 16\}$, siis $4 * 4 * 16 = 256$. Ka $\{6, 6, 7\}$ annaks kokku 252 võimalikku värvi (mida on veidi vähem kui lubatud). Erinevus nende kahe kombinatsiooni vahel $\{4, 4, 16\}$ ja $\{6, 6, 7\}$ on see, et esimene $\{4, 4, 16\}$ võimaldab esitada vähe punaseid ja rohelisi toone ning palju siniseid. Kombinatsioon $\{6, 6, 7\}$ on värvikanalite jaoks ühtlasema jaotusega. Et saavutada piiratud värvidega paremat tulemust, anna rohkem lubatud väärtuseid pildi dominantse tooniga sarnasele värvikanalile.

Teooria



Kaks lähenemisviisi 6 lubatud väärtuseni kvantiseerimisele

Valime pildi 256 värvini kvantiseerimiseks kombinatsiooni $\{6, 6, 7\}$. Et jaotada pidev väärtusvahemik 0 kuni 1 (näiteks punane värvikanal) kuueks lubatud väärtuseks, peame me konstrueerima eeskirja, mille alusel sisendväärtustest genereeritakse väljundväärtused. Me tahame kindlasti lisada väärtused 0 ja 1 ning nende vahele jätta neli vaheväärtust. Näidatud lahendus annab tulemuseks $\{0, 0.2, 0.4, 0.6, 0.8, 1\}$. 1.0 jagamisel 5-ga (vahede arv, võrdub lubatud väärtuste arv - 1) saame 0.2, mis annab meile teada, kui suured peaksid olema väärtuste vahed.

Et saada kenad ühtlased üleminekud, võime me väärtused jagada näiteks nii: 0.16 ja vähemale määrame vasteks 0.0; 0.16 kuni 0.33 määrame 0.2-ks; 0.33 kuni 0.5 kvantiseerime 0.4-ks jne ning 0.83 kuni 1.0 määrame 1.0-ks.

Funktsioon $f(x)$

Iga funktsioon koosneb matemaatilistest algeperatsioonidest (liida, lahuta, korruta, siinus jne) mille kombinatsioonide abil saame sisendväärtustest sobivad väljundväärtused.

Input		Function				Output	
8-bit	R value	*n	- 1/2	round()	/(n-1)	R value	8-bit
0	0.00	0.00	-0.50	0	0	0	0
13	0.05	0.30	-0.20	0	0	0	0
26	0.10	0.60	0.10	0	0	0	0
38	0.15	0.90	0.40	0	0	0	0
51	0.20	1.20	0.70	1	0.2	0.2	51
64	0.25	1.50	1.00	1	0.2	0.2	51
77	0.30	1.80	1.30	1	0.2	0.2	51
89	0.35	2.10	1.60	2	0.4	0.4	102
102	0.40	2.40	1.90	2	0.4	0.4	102
115	0.45	2.70	2.20	2	0.4	0.4	102
128	0.50	3.00	2.50	2	0.4	0.4	102
140	0.55	3.30	2.80	3	0.6	0.6	153
153	0.60	3.60	3.10	3	0.6	0.6	153
166	0.65	3.90	3.40	3	0.6	0.6	153
179	0.70	4.20	3.70	4	0.8	0.8	204
191	0.75	4.50	4.00	4	0.8	0.8	204
204	0.80	4.80	4.30	4	0.8	0.8	204
217	0.85	5.10	4.60	5	1	1	255
230	0.90	5.40	4.90	5	1	1	255
242	0.95	5.70	5.20	5	1	1	255
255	1.00	6.00	5.50	5	1	1	255

Antud juhul on funktsiooni põhielementideks korrutamine ja ümardamine/tüvenumbrite piiramine. Oletame näiteks, et me soovime konstrueerida funktsiooni, mis võtab sisendväärtuse vahemikus 0 kuni 1, näiteks 0.552, aga väljastab ainult väärtuseid 0.0, 0.2, 0.4 jne. Selleks peame me kõigepealt teisendama meie vahemiku 0 kuni 1 vahemikuks 0 kuni 6 ning lõikama komakoha tagused numbrid maha. Kuidas seda siis teha? Kõigepealt korrutame kõik väärtused 6-ga. Esimese *Math* sõlme väljundis on väärtused vahemikus 0 kuni 6. Et saada täisarve, peame me järgmiseks ümardama. Ümardame antud juhul täisarvuni (funktsioon *Round, Value* ehk numbrikoht milleni ümardada 1.0). Kuna me soovime 0-ks ümardada rohkem väärtusi, siis nihutame kogu väärtusvahemikku 0.5

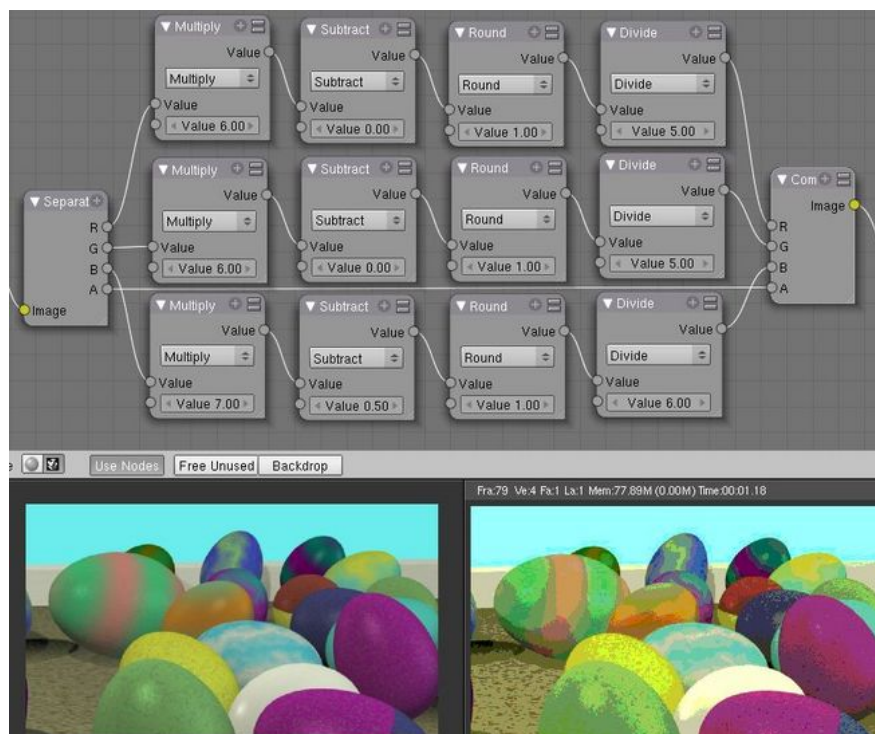
võrra negatiivses suunas (funktsioon *Subtract*). Ümardamisfunktsioon (*Round*) annab seejärel tulemused vahemikuks 0 kuni 5. Jagades tulemuse 5-ga, saamegi oma piksliväärtused jälle tagasi vahemikku 0 kuni 1. Kordame sama ka teiste värvikanalitega. Seega saime me tulemuseks funktsiooni

$$f(x,n)=\text{round}[x \cdot n - 1/2] / (n-1)$$

kus n on lubatud väljundväärtuste arv, x on piksli sisendväärtus ja $f(x,n)$ piksli väljundväärtus. Meil on veel vaid üks väike probleem. Nimelt väärtus 1.0 saab lõpuks väärtuseks 1.2, mis on väljaspool lubatud vahemikku. Sellise tulemuse saame me seetõttu, et ümardamisele minnes on tema väärtuseks 5.5, mis ümardatakse 6-ks. Vea parandamiseks võime me vahetulemusest lahutada hoopis 0.501, mispeale 5.499 ümardatakse kenasti 5-ks. Skaala teises otsas probleemi pole, sest musta värvi vaheväärtus -0.501 ümardub 0-ks, kuna funktsioon *Round* ei väljasta negatiivseid arve.

Sulle vajaliku tulemuse saamiseks on teinekord kasu taolisest funktsiooni lahtikirjutamisest, nagu selles näites. Siis on pilt selgem ja lihtsam ka sõlmekaardi kokku panna. Vaatame näitena juhtu, kus $n=6$ ja $x=0.70$. Leia tabelist rida, kus 8-bitine väärtus on 179 ja R väärtus 0.7 Korrutamine 6-ga annab tulemuseks 4.2. Lahutame 0.5 ja saame 3.7, mis ümardub 4-ks. 4 jagatud 5-ga = 0.8. Seega, $f(0.7, 6) = 0.8$ või 8-bitine väärtus 204. Tabelist on näha, et see sama 8-bitine väljundväärtus on mitme erineva sisendväärtuse tulemuseks. Tubli töö! Närisid ennast sellest läbi! Sedasi on võimalik matemaatikat Blenderi komposiitoris kasutada. Kui sa eelnevale pihta said, siis täna oma matemaatikaõpetajat.

Tegelikkus



Sellise operatsiooni korralikuks ülesehitamiseks toimi järgnevalt. Esiteks lahuta pilt *Separate RGB* sõlme abil eraldi värvikanaliteks. Punase värvikanali jaoks ühendame matemaatika sõlmed funktsiooniks, mis võtab iga punase tooni, korrutab selle väärtust soovitud jagamise hulgaga (6), vähendab tulemust 0.5 võrra, ümardab tulemuse lähima täisarvuni ning jagab seejärel pildipiksli värvi viiega. Ehk siis vahemikust $\{0 \dots 1\}$ saab $\{0 \dots 6\}$, lahutame 0.5 ja saame $\{-0.5 \dots 5.5\}$, ümardame ja saame $\{0, 1, 2, 3, 4, 5\}$, jagame 5-ga ja saame $\{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$.

Tulemuseks on värvikanali väljundi lubatud väärtuste piiratud arv. Kopeeri sõlmekaart ka rohelise ja sinise kanali jaoks. Et saada väärtuste arvu jaotust 6:6:7, säti RGB-kanalite jaoks korrutusteguriks vastavalt $\{6, 6, 7\}$ ja jagamisteguriks $\{5, 5, 6\}$.

Kui sa muudad kõik kasutatud sõlmed sõlmegrupiks, siis saad seda lihtsasti kasutada ka järgnevates projektides. Testimiseks võid kasutada sisendina mõnda erinevaid väärtusi genereerivat sõlme.

Kokkuvõtteks

Tavaliselt saame me renderdamise tulemusena pildi, mille värisügavus on 32 (RGBA) või 24 (RGB) bitti. Sellisel pildil saab esitada miljoneid erinevaid värvitoone. See sõlmekaart võtab taolise pildi ning vähendab värvide hulga 256 peale. Kui kanalid pärast töötlust jälle kokku kombineeritakse, on pildi iga värv vaid üks 256-st võimalikust.

Kuigi see sõlm kasutab kanalite eraldamiseks *Separate RGB* sõlme, võib kasutada ka teisi kanalite eraldamise ja kombineerimise variante. Kui sa kasutad YUV väärtusi (video kanalid), siis pea meeles, et U ja V komponendid on vahemikus -0.5 kuni 0.5, seega pead sa neile liitma 0.5, et saada vahemik 0.0-1.0.

JPG ja PNG pildiformaadid salvestavad värviinfot vastavalt formaadi standardsele bitisügavusele (näiteks JPG on 24-bitine).

Sa ei pea vähendama iga kanali värvide arvu ühtlaselt. Näiteks kui sinine on pildil domineeriv toon, siis võid sa jätta punastele kaks väärtust, rohelistele 4 ja sinistele 32 ning ikka mahuvad kõik värvid 256 sisse ära. Kasutades HSV komponente (toon, küllastus, heledus), võid vähendada küllastuse ja heleduse näiteks kahe väärtuse peale ning tooni 64 peale.

Seda meetodit võid kasutada ükskõik millise infokanali kvantiseerimiseks, olgu selleks siis alfa, kiirusvektorid, sügavus Z vms.

Kombineerimissõlmed (*Combine/Separate*)

Kõik need sõlmed teevad põhimõtteliselt sama asja: nad jagavad pildi eraldi kanaliteks või kombineerivad need kanalid jälle pildiks kokku. Kõik formaadid toetavad alfa (läbipaistvuse) kanalit. Värvide esitamiseks on mitmeid mooduseid (*color space* ehk värviruum). Blenderis on toetatud järgmised:

- RGB: *Red-Green-Blue* ehk punane-roheline-sinine, seda süsteemi kasutavad näiteks arvutimonitorid ja selles salvestatakse enamus pildifaile.
- HSV: kolm väärtust, mida mõned peavad intuitiivsemaks süsteemiks kui RGB:
 - *Hue*: toon;
 - *Saturation*: küllastus, tooni "värvilisus" vahemikus halltoon kuni väga erk värv;
 - *Value*: heledus, vahemikus 0.0 ehk must kuni 1.0 maksimaalse heledusega (kui küllastus on 0, siis valge).
- YUV: Heleduse ja värvinfo standard, mida kasutatakse PAL-videosüsteemis. Koosneb heleduse komponendist Y ja kahest värvikomponendist U ja V.
- YCbCr: Heleduse ja värvinfo standard, mida kasutatakse kõrglahutusega videos ja tele-edastuses (HDTV). Koosneb heleduse komponendist Y ja kahest värvikomponendist Cr ja Cb.

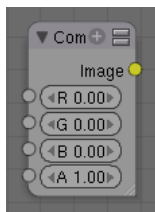
Värvisüsteemide kohta otsi rohkem infot internetist, näiteks Wikipediast.

Separate/Combine RGBA sõlm



Separate
RGBA sõlm

See sõlm eraldab pildist RGBA kanalid (punane, roheline, sinine ja alfa). Iga kanali jaoks on eraldi väljundpesa.



Combine
RGBA sõlm

See sõlm kombineerib eraldi RGBA kanalid üheks pildiks. Kasuta seda sõlme pärast kanalite eraldi töötlemist nende kokkuühendamiseks.

Näited



Selles näites võtame me alfakanali ja udustame seda, seejärel kombineerime ta jälle värvikanalitega kokku. Seda pilti mõne teise peale kombineerides näeme, et teravad servad on muutunud uduseks. See on peaaegu nagu sakisilumine. Võid seda kasutada näiteks renderdatud elementide kombineerimiseks filmitud materjaliga. Saad sellega muuta servad udusemaks ja seega oma materjali paremini kokku sobitada. Animeerides võid saavutada näiteks pilti sisse sulandumise efekti, kus objekt ilmub aeglaselt laienedes nähtavale.



Selles näites vahetame me pildi erinevad värvikanalid üksteisega ära ning saame naljakate värvidega pildi.

Separate/Combine HSVA sõlm



Separate
HSVA sõlm

See sõlm eraldab pildist HSVA kanalid (toon, küllastus, heledus ja alfa).

Kasuta eraldatud kanaleid vastavalt vajadustele. Näiteks võid heleduse (*V* ehk *Value*) kanalit korrutamistehtega skaleerides kogu pilti heledamaks/tumedamaks muuta. Võid muuta pilti pastelsemaks, vähendades tema küllastust (*S* ehk *Saturation* kanalit skaleerides).

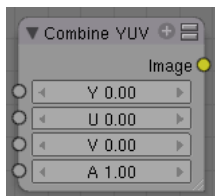
Võid ka eraldada mõne kindla värvitoonide vahemiku (näiteks lõigates *Hue* ehk tooni kanalit *ColorRamp* sõlme abil) ja muuta siis valitud värvide tooni.

Separate/Combine YUVA sõlm



Separate YUVA sõlm

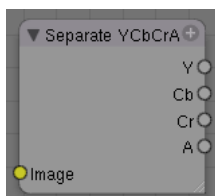
See sõlm eraldab pildist YUVA kanalid. Pane tähele, et U ja V kanalite info on vahemikus -0.5 kuni +0.5.



Combine YUVA sõlm

Kombineerib YUVA kanalid jälle üheks pildiks. Kui sa mõnda sisendpesa ei ühenda, võid selle väärtuse määrata numbrivälja abil.

Separate/Combine YCbCrA sõlm



Separate YCbCrA sõlm

See sõlm konverteerib RGBA pildi YCrCbA kanaliteks ning eraldab need:

- Y: heledus, 0=must, 1=valge
- Cb: sinise erinevuse värvikomponent, 0=sinine, 1=kollane
- Cr: punase erinevuse värvikomponent, 0=punane, 1=kollane

Märkus: kui sa kasutad nende kanalite töötlemisel *ColorRamp* sõlme, siis määra seal ülemineku tüübiks *Cardinal*, et saada parem tulemus. Heleduse kanali gamma (halltoonide teisendus) muutmisega on võimalik saavutada erinevaid kontrastiefekte.



Combine YCbCrA sõlm

Loodetavasti said pihta, millest eelnev jutt rääkis. Seega jääb järgneva sõlme funktsioon sulle endale nuputamiseks.

Alpha Convert sõlm

...

Komposiitimise maskimissõlmed (*Matte*)

Maskimissõlmede hulgast leiad erinevad tööriistad maskide loomiseks värvilistest (ingl. *chromakey*, *blue-/greenscreen*), harilikult sinistest või rohelistest taustadest. Ühevärvilisi taustu kasutatakse tavaliselt siis, kui on tarvis filmitavale materjalile hiljem midagi arvutiga juurde lisada, nii et lisatav jääks filmitu taha. Näiteks kui meil on filmitud näitleja, kes peaks olema kosmoselaevas ja taustal käib vinge kosmoselahing, siis on meil vaja näitleja kuidagi piiritleda, filmitud kaadrist eraldada, et ta siis meie loodud tausta peale kleepida. Seda piiritlemist nimetatakse tavaliselt maskimiseks ning seda on kõige lihtsam saavutada eelmainitud ühevärvilise tausta abil. Teised võimalikud tehnikad on rotoskoopimine (maskide joonistamine käsitsi) ning maski loomine mitte värviinfo, vaid mõne muu eristuse abil (näiteks heledus).

Maskisõlmede kasutamine pole alati nii lihtne kui teiste sõlmede kasutamine, seepärast on oluline katsetada ning leida seadete kruttimise kaudu parim kombinatsioon. Ka väikesed muutused võivad muuta pildi tunduvalt paremaks või halvemaks. Näiteks liigne tagaplaani värvieemaldus (inglise keeles kutsutakse seda *despill* või *unspill*) võib muuta su näitlejad näost plassiks või hoopis lillaks.

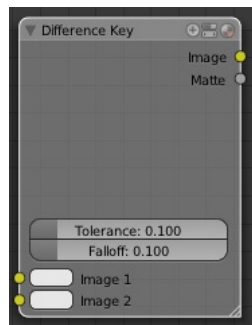
Maskimine pole enamasti ühe hiireklõpsu töö ning sageli on vaja erinevate sõlmede abi, et saada rahuldavat tulemust. Näiteks on vahel vaja luua eraldi maskid juustele või jalgadele ning need seejärel ülejäänuga kokku kombineerida.

Harilikult filmitakse nn *greenscreen*-võtteid studios, kus on kontrollitud valgustingimused. See võimaldab samade seadetega sõlmi kasutada väikeste muudatustega erinevate võtete puhul. Väliitingimuste puhul on valgus palju muutlikum ning ka samast suunast võetud võtete jaoks võivad sõlme seaded olla vägagi erinevad.

Prügimask

Prügimaskiks (*garbage matte*) kutsutakse käsitsi joonistatud rohmakat maski, mis lõikab välja suurema osa ebavajalikust taustast ja jätab edasiseks töötamiseks vaid kitsa ala meid huvitava objekti ümber. Näiteks kui meil on filmitud rohelisel taustal näitleja, kuid roheline taust ei ulatu päris kaadri servani või on kaadris statiivid, markerid või muud ebavajalikud elemendid, võime rohmaka maskiga eraldada vaid meid tegelikult huvitava ala (st näitleja koos kitsa osaga teda ümbritsevast taustast). Järgnevalt kasutame maski loomisel vaid välja lõigatud ala. See muudab maskimissõlmede töö kiiremaks ja täpsemaks ning tulemus saab parem.

Erinevusmask (*Difference Key*)



Erinevusmaski sõlm
Difference Key

Erienvusmaski sõlm määrab pildi iga piksli puhul tema erinevuse teisest pildist. Teiseks pildiks võib olla näiteks pilt rohelisest taustast ilma näitlejata või RGB-sõlme abil genereeritud ühtlast värvi taust. Kui erinevus kahe sisendpildi pikslite vahel on alla määratud läve, siis loetakse piksel taustaks ning muudetakse läbipaistvaks. Erinevusmask võimaldab maske luua suvalist värvi tausta järgi, kuid parima tulemuse saamiseks ei tohi esiplaani objektidel esineda taustaga sama värvi alasid. Näiteks kui taust on roheline, ei tohiks näitlejal olla seljas rohelist pluusi.

Sellel sõlmel on kaks sisendit.

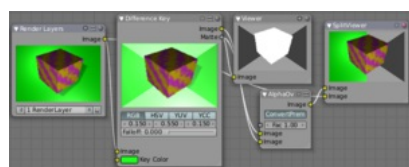
- Esimene ehk Image 1 on pilt, millele me maski tõmmata soovime.
- Image 2 on pilt, millega maskitava pildi piksliväärtuseid võrreldakse. Selle sisendina võib kasutada ükskõik millist värvilist pilti või ka näiteks RGB-sõlme abil genereeritud ühevärvilist pilti.

Sõlmel on kaks parameetrit. Esimene, Tolerance (lävi), määrab ära, kui suur võib olla maksimaalne erinevus kahe pildi pikslite vahel, et pikslit ikka veel taustaks loetaks. Mida suurem see on, seda rohkem sööb taust ennast pilti sisse. Teine parameeter, Falloff (üleminek), määrab selle, kui lai on üleminekuala tausta ja esiplaani vahel.

Kõik üleminekualale jääva erinevusega pikslid muudetakse poolläbipaistvaks. Kui parameetri Falloff väärtus on suurem, on üleminekud sujuvamad ning loomulikumad. Kui ülemineku (Falloff) väärtus on madal, on üleminek järsem ning üleminekud taustast esiplaaniks vähem sujuvad. Ülemineku väärtus tuleks määrata nii, et ta on maksimaalselt suur, kuid ei riku veel maski ära (ei lõika sisse esiplaani ega jäta esiplaani servi värviliseks).

Selle sõlme väljunditeks on pilt väljundpesast Image, mille alfakanalis on sõlme arvutatud piksite läbipaistvus, ning mustvalge pilt, nn alfamask väljundist Matte.

Lihtne näide

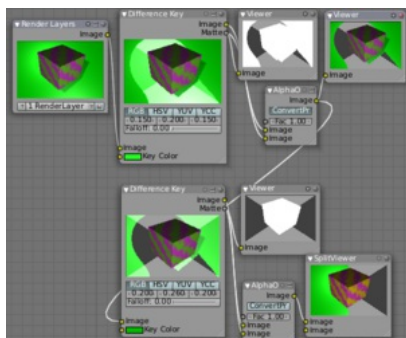


Erinevusmaski kasutamine

Meil on pilt lilla-kollase mustrilisest kuubist ebaühtlaselt valgustatud rohelisel taustal. Alustame sõlmekaardi ehitamist sellest, et lisame pildile erinevusmaski (*Difference Key*) sõlme ning valime pipeti abil pildilt helerohelise tooni kuubikule võimalikult lähedalt. Lisame kaks kuvamissõlme (*Viewer*), et näha, millises erinevusmaski väljundid. Lisame *AlphaOver* sõlme ning ühendame *Matte* väljundi selle **ülemisse** ühenduspessa ning pildi ehk *Image* väljundi **alumisse** ühenduspessa. Kumb kumba pessa on ühendatud, on väga oluline! Mask peab alati olema *AlphaOver* sõlme ülemises ühenduspessa. Lülita *AlphaOver* sõlmel sisse nupp *Premultiply* (eelkorutus). Ühendame *AlphaOver* sõlme väljundi võrdluskuva (*SplitViewer*) sõlme, et me saaksime võrrelda oma originaalpilti ning maski kasutamise tulemust. Ühenda originaalpilt võrdluskuva sõlme alumisse pessa.

Määrame erinevusmaski sõlme läveks (*Tolerance*) väärtuse 0.15 ning vaatame, mis me saame. Tulemuseks on mask, mis on kena küll ümber kuubi, kuid pildi servades on pikslierinevus nähtavasti suurem kui meie lävi ning roheline taust jääb paistma. Suurendame varieeruvust, et haarata sisse kogu roheline taust, kuid mitte esiplaani kuubikut. Katsetame erinevate väärtustega, kuni leiame rahuldava tulemuse. Kui kogu roheline on enam-vähem kadunud, on aeg timmida ülemineku laiust ehk parameetrit Falloff. Muuda seda nii, et ülemineku esiplaani servades oleks loomulik ja ühtlane, ilma et taust paistma hakkaks. Tulemuseks on taustast kenasti eraldatud kuup, mida võime kasutada edasistes komposiitmisoperatsioonides.

Mitme maskisõlme kooskasutus

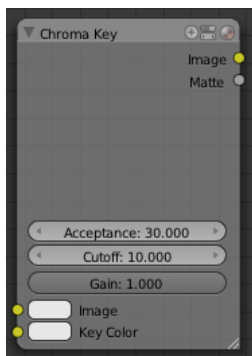


Erinevussõlmede koos kasutamine

Tihti peale ei piisa maski loomiseks vaid ühest sõlmest. Põhjuseid selleks võib olla mitmeid, näiteks on tausta varieeruvus liiga suur, et seda vaid ühe värvitooni ja lävega haarata. Selliste probleemide lahendamiseks on mõttekas jagada pilt tinglikult osadeks ning iga osa jaoks genereerida mask eraldi. Hiljem on juba lihtne kõik maskid omavahel kokku liita ning saada parem tulemus, kui ühe ainukeses sõlmega võimalik oleks. Selles näites on meil rohe-lilla mustriga kuup rohelisel ebaühtlasel taustal.

On selge, et ühe sõlmega me kuubi maski kätte ei saa, kuna me lõikaksime lisaks taustale välja ka osa kuubikust. Lahenduseks on kahe eraldi sõlme kasutamine, millest üks lõikab välja tausta heledama osa ning teine tumedama. Kuna kuubil olev roheline värv jääb nende lävedest väljapoole, siis seda ei puudutata ning me saame oma kuubi kenasti kätte.

Chroma Key sõlm



Chroma Key sõlm

Chroma Key sõlm on oma nime saanud vastavast tehnikast, millel on filmimaailmas pikk ja viljakas ajalugu. Tegemist on analoogfilminduse aegadest pärit terminiga, kus ühe fotoemulsiooni kihiga sama värvi tausta sai fotolaboris kavalate nippidega eemaldada ning asendada mõne teise taustaga. Arvutiajastul pole tausta värv enam nii oluline, kuid traditsioonilised sinine ja roheline värv on kõige laiemalt kasutatavad. Roheline omab eelist just digitaalmaailmas ning video puhul, kuna roheline värv on kõige erksam ning säilitab enamasti videosignaalis kõige rohkem oma infost.

Sõlmel on üks sisend, Image ehk pilt, millele sa soovid maski luua.

Sõlme parameetriteks on:

Acceptance ehk põhimõtteliselt tolerant

See määrab, kui suures osas võib tausta värv määratust varieeruda, et teda ikka veel taustaks loetaks.

Cutoff - üleminekuuala laius, sama mis Falloff teistes sõlmedes

Määrab, millises väärtuste vahemiku ulatuses tolerantsi piiridest edasi tõlgendatakse tausta poolläbipaistvana.

Gain (võimendus)

Loodud maski skalaarkordaja, võimaldab muuta maski "tihedust" ning säilitada maski poolläbipaistvaid alasid, näiteks varje.

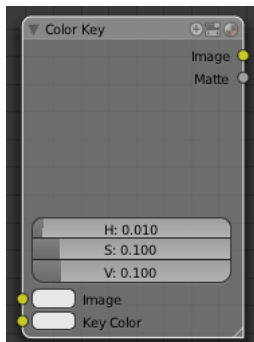
Chroma Key sõlm on disainitud spetsiaalselt ühevärvi taustade pealt maskide loomiseks. Sõlm töötab kõige paremini, kui tausta värviks on üks RGB primaarvärvidest, st roheline, sinine või punane.

Video puhul kasutatakse enamasti rohelist tausta, kuna roheline on kõige heledam ning on videosignaalis kõige inforohkem värv.

Sinine värv on traditsiooniliselt olnud kasutuses filmi puhul, kuna sinine on nahatoonidele rohkem vastanduv ja sobib seega inimeste filmimiseks paremini.

Selle sõlme väljunditeks on pilt väljundpesast Image, mille alfakanalis on sõlme arvutatud piksite läbipaistvus, ning mustvalge pilt, nn alfamask väljundist Matte.

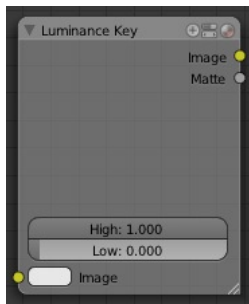
Värvimaski sõlm (*Color Key*)



Color Key sõlm

Värvimaski sõlm võimaldab maski loomist samuti määratud värvi alusel, kuid tema seadete kaudu on võimalik määrata, kui suur võib olla tausta värvi erinevus eraldi tooni, küllastuse ja heleduse osas. Sõlme kolme liuguri abil saab kontrollida eraldi tooni (Hue), küllastuse (Saturation) ja heleduse (Value) lubatud varieeruvuse piire. Suuremad väärtused tähendavad, et varieeruvus võib olla suurem.

Heledusmaski sõlm (*Luminance Key*)



Luminance Key sõlm

Heledusmask (Luminance Key) võimaldab maski loomiseks kasutada pildi piksliväärtuste heleduse infot. Põhimõtteliselt on võimalik sama saavutada ka värvimaski abil, seades seal tooni varieeruvuse maksimaalseks ning küllastuse ja heleduse omad vastavalt vajadustele. Heleduse abil maski loomine on kasulik, kui meil on tumedad objektid heledal taustal (näiteks lind heleda taeva taustal) või hele objekt tumedal taustal (kasutatakse tihti eriefektide lisaelementide, näiteks tule ja plahvatuste filmimiseks).

Sõlmel on üks sisend, Image ehk pilt, millele sa soovid maski luua.

Sõlme parameetriteks on:

- High (kõrge) suurim piksliväärtus, mida veel esiplaaniks loetakse. Ehk esiplaani heledus võiks olla vahemikus "High" kuni 1.0.
- Low (madal) madalaim piksliväärtus, mida veel taustaks loetakse. Peaks olema väiksem kui *High* ja suurem kui 0.0.

Suurima ja vähima vahele jääv ala on üleminekuala, kus pikslid muudetakse poolläbipaistvaks.

Selle sõlme väljunditeks on pilt väljundpesast Image, mille alfakanalis on sõlme arvutatud piksite läbipaistvus, ning mustvalge pilt, nn alfamask väljundist Matte.

Näide



Heledusmaski kasutamine koos maski pööramisega

Selles näites on meil modell vaja paigutada teisele taustale. Modelli filimiti "valgel" taustal (mis pole just täiuslik valge, aga käib kah). Heledusmaski sõlme abil saame me maski, mis on läbipaistev seal, kus on modell, ning läbipaistmatu seal, kus on taust. Seda me loomulikult ei taha. Asja parandamiseks peame me maski ümber pöörama, musta ja valge omavahel ära vahetama. Kuidas seda teha? *ColorRamp* sõlm päästab meid. Määrame ülemineku vasakpoolse otsa valgeks värviks ning parempoolse mustaks. See pöörab meie maski kenasti vastupidiseks, seal, kus oli must, on valge jne. Joonisel on kujutatud pööratud mask, kus valge tähistab modelli nagu peab.

Modelli taustale paigutamiseks kleebime ta segamissõlme (*Mix*) abil tausta peale ning ühendame *Matte* väljundi segamissõlme faktori sisendisse. Üks ja veel lihtsam variant on maski mitte pöörata ning vahetada lihtsalt segamissõlme sisendid omavahel ära. Sisendite vahetamine paigutab tausta hoopis modelli peale, kuid teeb seda seal, kus modelli ei ole ehk täpselt meile vajalikes kohtades. Sedasi saame maski pööramise vahelt ära jätta ning tulemus on sama. Meie modell on teleporteerunud otse kruisilaeva

dekile!

Värvilekke eemaldaja (*Color Spill*)



Color Spill sõlm

Värvileke on nähtus, kus tausta värv "lekib" esiplaanile, määrides esiplaani servad ja toonides neid tausta värvidega. Värvilekke (Color Spill) sõlm on mõeldud selle probleemi vähendamiseks (päriselt eemaldada ta seda siiski ei suuda). Töötab ta väga lihtsal põhimõttel, nimelt kanalite piiramise abil. Näiteks kui meil on tausta värviks roheline, piiratakse esiplaani pikslite rohelist värvikanalit kas punase või sinise kanali tasemeni, vastavalt sellele, kumb on kõrgem. See muudab piksli tooni vähem rohekaks, kuid ei suuda siiski taastada tema esialgset värvi. Värvileket saab vähendada, kuid seda täielikult ära hoida on sageli keeruline. Põhjusteks on nii esiplaani objektide läiked (peegeldavad tausta), taustalt esiplaanile peegeldunud tausta värvi valgus kui ka video- või filmikaamera optika kvaliteedist põhjustatud värvide hajumine. Värvileke on seda tugevamalt nähtav, mida heledam on esiplaan. Värvilekke visuaalseks vähendamiseks kasuta seda sõlme.

Sõlmel on kaks sisendit - Image ehk pilt, millelt värvileket eemaldada tahetakse, ning Fac ehk faktor, mis võimaldab kontrollida efekti tugevust. Seda läheb vaja näiteks siis, kui meil on vaid kindel ala, mida me töödelda tahame.

Valik Despill Channel (lekkekanal) määrab, millist värvikanalit me maha suruda tahame. Kuna sõlm töötab põhimõtteliselt kanalite limiteerijana, siis ei saa siin määrata suvalist värvi. Valik on üks kolmest RGB kanalist. Algoritmi valiku hüpikmenüüst saad valida, kas piiramine on lihtne (*Simple*) või ülejäänud kanalite keskväärtuse (*Average*) alusel toimiv. Valik Limiting Channel (piirav kanal) on kanal, mille väärtustega piiravat kanalit võrreldakse ning milleni see piiratakse. Valik Ratio (suhe) võimaldab kanalite omavahelist suhet skaleerida, sisuliselt määrab see värvilekke eemaldamise mõju. Valiku Unspill (lekke-eemaldus) valiku sisselülitamisel avanevad lisavõimalused kõigi kolme kanali eraldi kompenseerimiseks.

Sõlmel on vaid üks väljund, mille kaudu saame kätte töödeldud pildi.

Kanalimask (*Channel Key*)



Channel Key sõlm

Kanalimask ehk Channel Key loob maski määratud värvikanali piksliväärtuste alusel. Valida saab vastavalt vajadusele erinevate kanalite kombinatsioonide vahel. Näiteks YUV värviruumi Y-kanal kui peamist heledusinfot sisaldav oleks kasulik heledate objektide eraldamiseks tumedast taustast.

Sõlmel on üks sisend, Image ehk pilt, millele sa soovid maski luua.

Sõlme parameetriteks on:

- Color Space värviruum, mille kanalit sa kasutada tahad.
- Channel kanal, mille väärtuste alusel mask luuakse.
- Algorithm kasutatav algoritm. Kas lihtsalt ühe kanali väärtused või kanalite maksimum. *High esiplaani heleduse vähim väärtus.
- Low tagaplaani suurim väärtus. Ära lase ennast häirida sellest, et esiplaan on sinu materjalis tumedam kui tagaplaan. Alati on võimalik mask hiljem ümber pöörata, st must valgeks ja vastupidi.

Suurima ja vähima vahele jääv ala on üleminekuala, kus pikslid muudetakse poolläbipaistvaks.

Selle sõlme väljunditeks on pilt väljundpesast Image, mille alfakanalis on sõlme arvutatud piksite läbipaistvus, ning mustvalge pilt, nn alfamask väljundist Matte.

Kaugusmask (*Distance Key*)

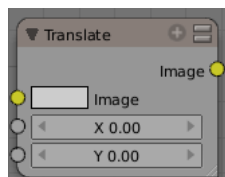
Kaugusmask loob maski, arvutades pildi iga piksli jaoks tema "kauguse" tausta värvist. Kui kujutleda RGB värviruumi kolmeteljelise ruumina, siis saab iga piksliväärtuse sinna kindlasse kohta paigutada. Kahe erineva piksliväärtuse vahe on võimalik välja arvutada ning selle vahe alusel loobki antud sõlm maski. Sõlmel on kaks parameetrit, Tolerance (lävi) määrab kauguse värviruumis, millest

lähemale jäävad pikslid loetakse veel taustaks ning Falloff (nõrgenemine) määrab taustast esiplaaniks ülemineku ala laiuse ehk poolläbipaistvuse ulatuse. Mida väiksem see on, seda teravam on mask ning seda sakilisemad on maski servad.

Komposiitimise moonutussõlmed (*Distort*)

Need sõlmed moonutavad ja liigutavad pilti kas kogu pildi ulatuses korraga või vastavalt määratud maskile.

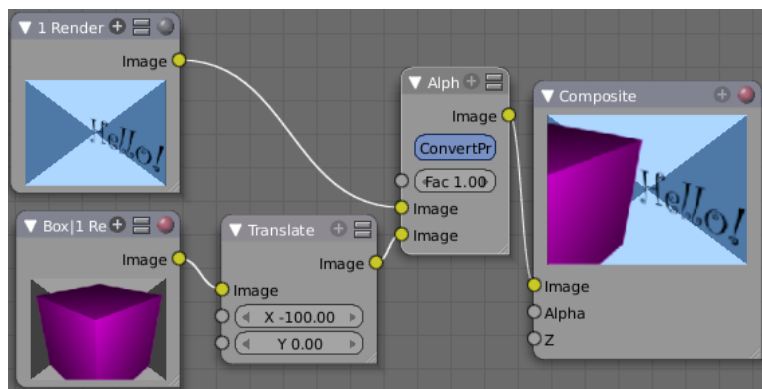
Nihutussõlm (*Translate*)



Nihutussõlm Translate

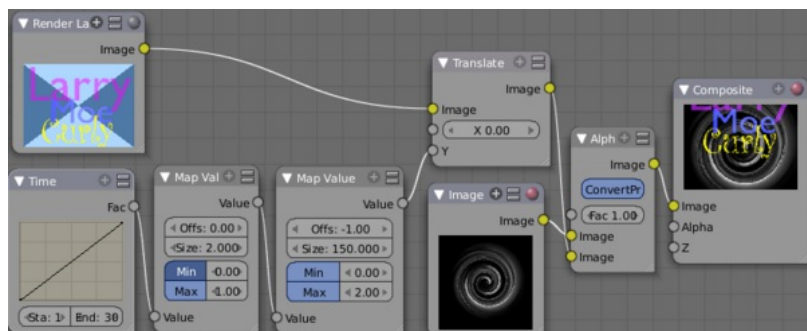
Nihutussõlm liigutab pilti määratud ulatuses kas X- või Y-telje suunas. X- ja Y-ühikuks on pikslid ja väärtused võivad olla nii positiivsed kui ka negatiivsed. X on pildi horisontaaltelg ja Y vertikaaltelg ning telgede ristumispunkt on pildi vasakpoolne alumine nurk. Et liigutada pilti näiteks üles ja vasakule, tuleks määrata nihkeks (*Offset*) X-teljel negatiivne väärtus ja Y-teljel positiivne.

Kasuta seda sõlme elemendi paigutamiseks komposiitpildi loomisel. Harilikult on selle sõlme väljund sisendiks kas *AlphaOver* või *Mix* sõlmele komposiitpildi loomiseks. Alltoodud näites on renderduskihi pilt (kuup) nihutatud vasakule (negatiivne nihe X-teljel) ja kombineeritud teksti kujutava renderduskihiga.



Näide: kuidas kasutada nihutussõlme (*Translate*) kerivate tiitrite loomiseks

Kui sa soovid oma animatsiooni lõppu lisada kenad kerivad tiitrid, siis loe edasi. Kerivad tiitrid on sellised, kus tekst liigub ekraani ühest servast (tavaliselt alumisest) sisse ning teisest servast välja. Allolev joonis näitab, kuidas luua nihutussõlme abil kerivate tiitrite sõlmekaart.



Nihutussõlme kasutamine kerivate tiitrite loomiseks

Renderduskihil on meil inimeste nimed, kes selle kena animatsiooni loomisel osalesid. Seda pilti kasutame me nihutussõlme sisendina. Nihutussõlme Y-telje nihke väärtuse genereerime me ajasõlmega ning määrame, et nihke väärtus muutub vahemikus -150 kuni 150 30 kaadri jooksul. Nihutatava pildi paigutame keerise pildiga tausta peale. Tulemusena kerib meie tiitrite pilt 30 kaadri jooksul üle ekraani alaservast ülaservani. Kuna kirjadega pildi kõrgus on 150 pikslit, siis alguses ja lõpus on ta kenasti "serva taga". Loodud tiitrite nägemiseks tuleb meie kompositsioon renderdada animatsioonina. Antud hetkel, kaadris 21, on ekraanil nimed Moe ja Curly, Larry on juba ekraani ülaserva taha kadunud.

Alfakanal

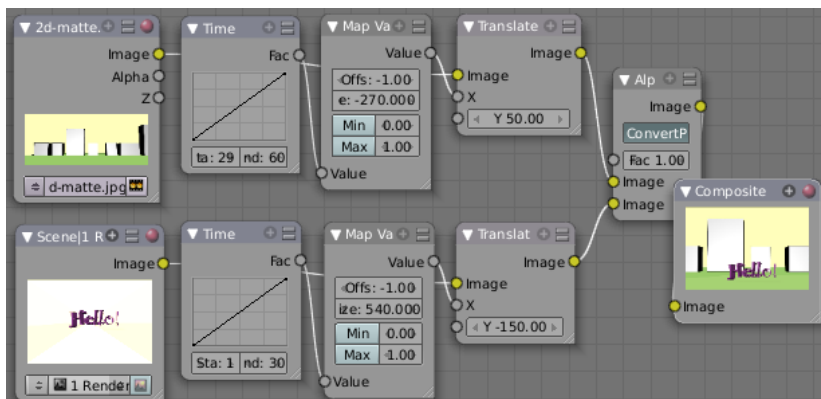
Ära unusta salvestada oma tiitrite tekstide pildifaile formaadis, mis lubab ka alfakanalit (näiteks PNG), sest vastasel juhul ei saa sa neid muu tausta peale kombineerida.

Sa võid ju kogu teksti ka kaaderhaaval otse 3D-kaamerast välja renderdada, kuid see võtab üüratu aja ja pole nii paindlik kui tiitrite lisamine järeltöötluses. Nihutussõlme (*Translate*) kasutamine on palju kiirem ja mugavam. Et kellegi nime lisada või eemaldada, muuda lihtsalt nimesid sisaldavat pildifaili. Kuna tegemist on lihtsa pildimanipulatsiooniga, on see operatsioon Blenderis väga kiire. Samuti on sel viisil väga lihtne muuta tausta - lihtsalt asenda komposiitsõlmes taust, renderda tiitrid uuesti välja ja ongi valmis.

Näide: animeeritud taust

Mõnedes 2D- ja 3D-animatsioonides ja filmides kasutatakse taustana kas joonistatud või renderdatud pilti (*matte painting*). Kui kaamera ei liigu, on taust paigal, kuid kui kaamera pöörab, siis saad tausta animeerida näiteks nihutussõlme (*Translate*) abil. Taustapildid on enamasti suuremad kui lõppkaader, antud juhul näiteks 1440×600 pikslit, kuigi lõppkaadri suurus on standardne telekaader. Suurem taust annab meile võimaluse seda ringi liigutada. Selles näites on meil esiplaani elemendiks tantsiv "Hello!"

tekst, mis liigub kaadris vasakult keskele. Kui tekst jõuab kaadri keskele, lisame me simuleeritud kaamera liikumise, liigutades tausta vasakule. See jätab mulje, nagu liiguks tekst jätkuvalt edasi ning kaamera jälgiks seda.



Animeeritud taust animeeritud elemendiga

Kasuta *Map Value* sõlme, et skaleerida X-koordinaadi nihked nihutamissõlme (*Translate*) jaoks. Pane tähele, et nihkeväärtusi *Offsets* on kasutatud ka teksti nihutamiseks madalamale, kui ta originaalrenderdusel on, et jääks mulje nagu ta jalutaks mööda tänavat (renderdatud stseenis on see kaamera keskel ja lihtsalt tantsib kohapeal). Samuti on taustapilti nihutatud ülespoole, et tausta ja esiplaani paremini kokku sobitada.

Näide: kaamera rappumine

Üks efekt, mida on vahetevahel vaja simuleerida, on kaamera rappumine. PÕMM! Me ootame, et nähtud plahvatus oleks reaalselt "tunda". Virtuaalmaailmas aga oleme me vaakumis ja efektid kaamerat ei mõjuta. Seega peame me kaamera rappumise ise simuleerima. Seda võib teha kahel erineval viisil, kas otse *animatsiooni* kaamera teisendust muutes või siis *järeltöötluses*.

Animatsiooni loomisel võime me kaamerat lpo-kõverate ja seda pööravate ja/või liigutavate võtmekaadritega niimoodi animeerida, et ta õigel hetkel sobivalt rapuks. Selle eelis on, et tulemus on realistlikum (näiteks on sedasi võimalikud ka rappumisest tulenevad perspektiivimuutused) ning animatsiooni kokku monteerides ei pea enam rappumise pärast muretsema, kuna see on juba tehtud. Puuduseks on see, et vastav kunstnik võib tegeleda ainult näitlejate modelleerimisega ning seetõttu ei mõjuta kaamera rappumine tausta, maastikut ega rekvisiite. Seega on mõttekas lisada kaamera rappumine järeltöötluses, kui kõik on juba kenasti välja renderdatud ja kõik elemendid kombineeritud.

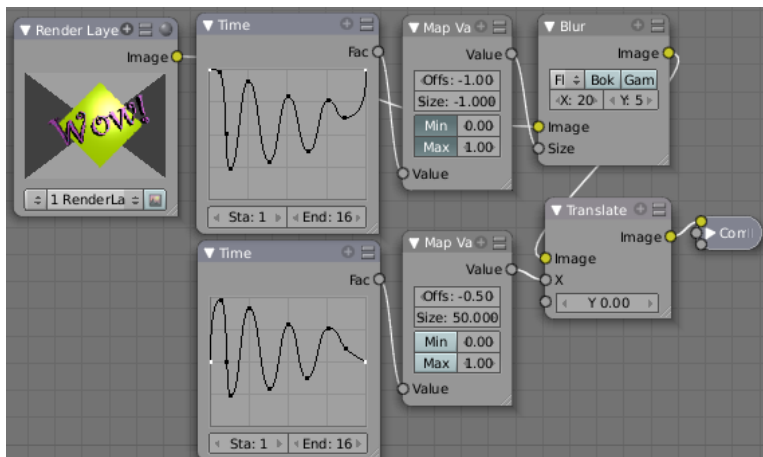
Kaamera rappumine, tuleneb see siis kas mõnest müksust, statiivi otsa koperdamisest või plahvatuses, mis sinu kõrval just aset leidis, koosneb kahest eraldi komponendist. Nendeks on pildi liikumine ja liikumishägu.

Pilt **liigub**, kuna kaamera filmimise ajal reaalselt liikus. Kaamera liikumist mõjutavad muidugi selle mass ja ka statiiv, millel kaamera seisis, summutades otseseid lööke. Ka operaator mõjutab oma kehaga kaamerat ning summutab ebasoovitavaid liikumisi korrigeerides pidevalt kaamera asendit.

Kaamera liikumise ja algasendisse naasmise vahel võib olla ka päris pikk paus, pikem paus annab näiteks aimu, et meie virtuaalsel operaatoril võttis kõmakast toimumine veidi aega. Kõik sõltub sellest, millist efekti sa soovid saavutada ja millist lugu jutustada.

Liikumishägu tekib sel põhjusel, et iga kaader salvestab mitte hetke, vaid kindla ajavahemiku jooksul juhtunu. Enamasti on liikumised ühe kaadri ajal väikesed ning liikumishägu seetõttu vähemärgatav. Kiirete liikumiste puhul aga on kogu pilt udune ning udu "suund" on liikumise suunaline. Liikumishägu hulk sõltub seega otseselt kaamera ja nähtavate objektide liikumise kiirusest iga kaadri ajal. Maksimaalse kõrvalekalde momendil on liikumishägu vähem kui kõrvalekalde algus- ja lõppfaasis, sest kaamera seisis sel hetkel enam-vähem paigal. Kaamera rappumise puhul on rohkem udu alguses ja vähem lõpus, kui kaamera muutub jälle juhitavaks. Lisaks võib kaamera rappumise ajal minna fookusest välja ning pärast rappumise lõppu jälle fokuseerida.

Blenderi sõlmede abil kaamera rappumise imiteerimiseks võid kasutada alltoodud skeemi. Siin on meil pildi udustamise sõlmede grupp (*Blur*) ja pildi nihutamise sõlmede grupp (*Translate*), kusjuures udustamisgrupi väljund on nihutamise sisendiks.



Eriefekt: kaamera rappumine

Eeltoodud näites kasutame me ajasõlme (*Time*) ajakõvera, et imiteerida raputuse intensiivsust ja kestust. Mõlemal ajakõveral on neli tippu 16 kaadri jooksul, mis imiteerivad raputamist. Kõige mõttekam on kõigepealt üks kõver valmis teha ning seda seejärel teda kopeerida ja vastavalt vajadusele veidi muuta. Pane tähele, kuidas kõvera võnkeamplituud lõpuks sumbub. Sumbumine annab edasi kas raputava nähtuse vaibumist või ka seda, kuidas operaator kaamera üle jälle kontrolli tagasi saab. Pane tähele, et kõver on siinusfunktsiooni sarnane, kuna rappumise käigus liigub kaamera alqpositsioonist nii ühele kui ka teisele poole.

PÕMM! Nihke algväärtus on kõveral 0.5. Esimese, kõige tugevama nihke korral liigub kaamera nii palju kui võimalik, kõvera maksimumväärtuseni, samas teisele poole liikudes on amplituud väiksem. See annab edasi operaatori vastupanu raputuse mõjule ja esimest kõige suuremat pauku, mis kaamera raputuse keskme nii-öelda paigast ära löi. Seega liigub kaamera alguses rohkem vasakule ning jääb seejärel uue keskme ümber rappuma.

Liikumine ja udu on tekitatud sama mõjutuse poolt, kuid nende kõverad peavad olema erinevad. Nihke maksimaalse amplituudi ajal peab udu olema null, kuna sel hetkel kaamera praktiliselt seisab (hakkab kohe-kohe tagasi liikuma). Selleks on udustamise ajakõvera algus- ja lõpp-punkt paigutatud teise kohta kui nihke ajakõveral. *Map Value* sõlme abil on ajakõvera väljundid skaleeritud nihke ja udustamise sõlmede jaoks sobivaks.

Kasuta joonisel näidatud sõlmekaarti kaamera rappumise imiteerimiseks. Rappumise amplituudi ulatus on määratud *Size* väärtusega nihke *Map Value* sõlmes. Udustamine peaks olema nihke suuna ja tugevusega proportsioonis. Kasuta ajasõlme (*Time*) algus- ja lõppväärtusi, et muuta raputamise kestust. Häid näiteid kaamera raputamise kasutamisest võib saada filmidest ja erinevatest animatsioonidest.

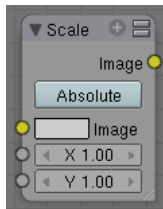
Pööramissõlm (*Rotate*)



Pööramissõlm
Rotate

See sõlm pöörab pilti. Positiivsed väärtused pööravad pilti päripäeva ja negatiivsed vastupäeva.

Mõõtkava teisendamise sõlm (*Scale*)



Mõõtkava
teisendamise
sõlm Scale

See sõlm suurendab või vähendab pilti. Mõõtkava teisendamine võib olla kas absoluutne või suhteline. Kui nupp *Absolute* on sisse lülitatud, saab määrata pildi uue suuruse pikslites. Suhtelise mõõtkava teisendamise puhul saab uue suuruse anda protsendina originaalist.

Näiteks kui X: 0.50 ja Y: 0.50, siis on väljundpildi mõõtmed poole väiksemad kui sisendpildil. Pildi tugeval suurendamisel on mõttekas seda veidi udustada, kuna muidu jäävad suurendatud pikslite kandilised servad väga näha. See võib muidugi olla ka taotluslik.

Kasuta seda sõlme, et erinevate elementide suuruseid ühtlustada. Enamik sõlmi genereerib väljundina pildi, mis on sama suur kui sisendpilt. Seega, kui sa tahad kombineerida kahte erineva suurusega pilti, pead sa nende suurused ühtlustama.

Peegeldamissõlm (*Flip*)



Peegeldamissõlm
Flip

See sõlm võimaldab pilti peegeldada kas X- või Y-telje suunas või mõlemas korraga.

Seda sõlme võid kasutada ka päris peegli efekti loomiseks. Selleks kombineeri peegeldatud pilt originaalpildiga näiteks poolitava maski abil.

Pikslinihutussõlm (*Displace*)

Oled sa kunagi jälginud asfaltteed kuumal suvepäeval? Ja pannud tähele, kuidas taust asfaldi kohal kuumas õhus virvendab? See toimub seetõttu, et erineva temperatuuriga õhk murrab valgust erinevalt ning tekitab omapärane lainetav pilt. See sõlm võimaldab luua samasuguseid efekte. Ta nihutab sisendpildi piksleid määratud vektormaski abil. Vektormaski väärtused määravad, kui palju sisendpildi igat pikslit tuleb nihutada.

Pikslinihutust saab kasutada mitmete erinevate efektide loomiseks, näiteks eelmainitud kuum õhk, valguse murdumine läbi läätse või muu läbipaistva objekti jne! Mäletad, kuidas superarvuti HAL filmis "2001: Kosmoseodüsseia" nägi maailma läbi moonutava lainurkobjektiivi? Seda efekti annab simuleerida pikslinihutussõlme abil.

Nihke ulatust ja suunda saab kontrollida:

- Vektormaski piksliväärtustega:

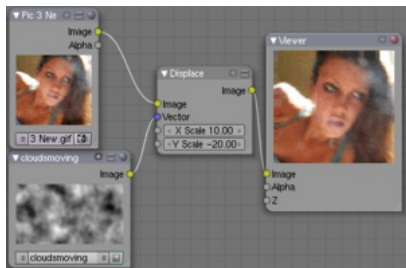
- Vektormaski mõõtkava teisendamisega

Punase kanali väärtus määrab nihke ulatuse X-telje suunas, roheline aga Y-telje suunas.

Kui piksliväärtused punases ja rohelises kanalis on võrdsed, siis nihutatakse sisendpildi pikslit mõlemal teljel sama palju. Nihke ulatust mõlemal teljel saab kontrollida ka sõlme X ja Y mõõtkava teisendamise väärtuste abil. Mõõtkava teisendamise väärtusi kasutatakse vektormaski piksliväärtuste mõõtkava muutmiseks (korrumamiseks), et muuta nihke mõju tugevust vastava telje suhtes. Mõõtkava teisendamise väärtus peab olema nullist erinev, et sellel mingit efekti oleks (nulliga korrumamine annab ju tulemuseks nulli).

Pikslinihutussõlme annab kasutada kahel erineval viisil: nii ise käsitsi joonistatud maski kasutades kui ka mõnel kavalal viisil genereeritud pildi abil (näiteks pinnanormaale või liikumisvektoreid sisaldavad pildid).

Näide



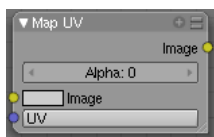
Muusikavideoliku efekti loomine pikslinihutuse abil

Selles näites laulab pildil olev neiu oma unistustest. Unenäolisuse rõhutamiseks kasutame animeeritud pilvetekstuuri pikslinihutuse vektormaskina. Kuna pilvetekstuuri piksliväärtused on vahemikus 0 kuni 1, siis peame me nende mõõtkava teisendama, vastasel juhul oleks maksimaalne võimalik nihe vaid üks piksel. Seega määrame me X- ja Y-telje mõõtkava teisenduseks sobivad väärtused (näites vastavalt 10 ja -20).

Kuna võrdne nihe X- ja Y-teljel muutis neiu näost veidi paksukeseks, siis on teisendustegur X-telje jaoks väiksem kui Y-telje jaoks. Nüüd näeb ta nagu välja veidi kõhnem ja pikem. Selline lihtne sõlmekaart ei anna küll kõige edevamat tulemust, kuid efekti põhiolemus on siit kenasti näha.

Fotode autor on Becca, õigused pole reserveeritud. Et näha mõningaid animatsioone, kus seda sõlme on kasutatud, vaata [silia](#)

Map UV sõlm



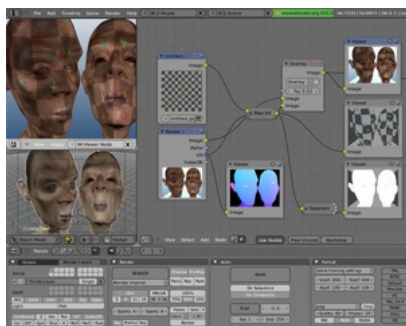
Tihtipeale on inimeste probleemiks see, et nad ei tea, mida nad tahavad. Sama lugu on ka režissööridega. Kuigi me oleme andnud oma parima meie loodud mudeleid tekstuurides ja animeerides, muudab režissöör tihtipeale meelt. "Ma tahaks, et ta näeks välja veidi karsem. Kes selle meigi talle õigupoolest tegi?" kõlab küsimus. Kuigi järeltöötuses saab teha päris palju asju, on ka sellel omad piirid. See sõlm annab sulle aga võimaluse objekte **ümber tekstuurida "pärast" seda, kui nad on välja renderdatud**. Kõlab segaselt, kas pole? Aga nii see on.

Selle sõlme abil (ning salvestades UV-laotuse näiteks OpenEXR-i pildifaili kihina) saad sa muuta objektide teksture. Eriti mugav on seda teha siis, kui sa oled [ID Mask sõlme](#) abil salvestanud ka objektide unikaalsete ID-väärtuste kihi.

Ühenda uus tekstuur pildi sisendisse (*Image*) ning UV-laotuse pilt UV-sisendisse. Tulemuseks on pilt, kus sisendtekstuur on väanatud vastavalt UV-koordinaatidele. Seda pilti saab nüüd kombineerida mõne teise pildiga, näiteks sellisega, kus ümbertekstuurimist vajav objekt kujutatud on. Alfa ja segamiskorduri abil saad *Mix* sõlmes kontrollida, kui palju katab uus genereeritud pilt vana.

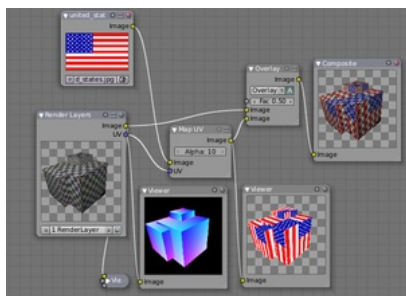
Uue tekstuuri loomisel on objekti originaaltekstuur loomulikult abiks, seega hoia see käepärast.

Näited



Ruudustiku lisamine objektile kaamera liikumise tuvastamise lihtsustamiseks

Selles näites lisame me ruudustiku kahe renderdatud pea animatsioonile. Renderdamiseks lülitasime me sisse UV-kihi salvestamise (nuppude aken *Buttons*, renderduskontekst *Render*, renderduskihi sakk *RenderLayer*). Segamissõlme *Mix* abil lisame me ruudustiku originaalpildi nägudele. See ruudustik on meile nüüd abiks, kui me peame renderdatud pildil tegema mingisugust liikumise jälitust.



UV-tekstuuri lisamine järeltöötluses

Selles näites lisame me lipu tekstuuri väljarenderdatud kandilisele objektile. *Mix* sõlme seadetes tuleb sisse lülitada ka alfa korrutamise määrang (*Pre-Multiply*) Lipu tekstuuri kasutame me lisatekstuurina objekti olemasoleva tekstuuri peal. Seda operatsiooni võib kasutada näiteks sildivahetusteks erinevatel objektidel (pesupulbripakk reklaamis jne).

Loomulikult pole sellest sõlmest režissöörile rääkimine kõige parem mõte, kuna siis hakkab ta arvama, et järeltöötlus on kõikvõimas (mida see loomulikult ongi). Seega hoiame selle väikese saladuse esialgu enda teada.

Lõikamissõlm (*Crop*)

Crop sõlm lõikab pildist välja määratud ala.

Crop Image Size (lõika pildi suurust)

Selle sisse lülitamisel muudetakse ka väljundpildi mõõtmeid. Väljalülitatult muudetakse serva taha jäävad pikslid lihtsalt läbipaistvaks ja pildi mõõtmed ei muutu.

Relative (suhteline)

Kui see on sisse lülitatud, määratakse lõigatav ala protsendina originaalpildi mõõtmetest. Väljalülitatult saame lõigatava ala suuruse määrata pikslites.

Crop Region Values (lõikeala piirid)

Need väärtused määravad lõigatava ala ulatuse alt, ülalt, vasakult ja paremalt.

Läätsemoonutuse sõlm (*Lens Distortion*)

Kasuta seda sõlme kaamera objektiivi moonutuste simuleerimiseks.

Distort (moonutus)

See määrab moonutuse mõju tugevuse ja tüübi - kas tünni- või padjakujuline.

Dispersion (dispersioon)

Simuleerib erineva lainepikkusega valguse erinevast murdumisest tulenevat vikerkaareefekti pildi servaaladel. Efekt on seda tugevam, mida rohkem serva poole.

Projector (projektor)

Projektori tüüpi moonutus. Sisselülitatult tekitatakse moonutus vaid horisontaalsuunas. Sisselülitatult tühistab see seadete Jitter (võbin) ja Fit (sobitus) valikud.

Jitter (võbin)

Lisab moonutusele müra. Kiirem, kuid mürasem.

Fit (sobitus)

Suurendab pilti, et tekkivad mustad alad näha ei jääks. Töötab vaid moonutuse positiivsete väärtuste korral.

Monteerimine

Lisaks modelleerimisele ja animeerimisele sisaldab Blender ka kõigi vajalike võimalustega videomonteerijat (Video Sequence Editor, VSE) ning sõlmedel baseeruvat keerukat ja rohkemate võimalustega video- ja pildikomposiitorit. [Komposiitimisõlmed](#) töötavad ühtmoodi hästi nii piltide kui videoga ning nende abil saab pilte/videosid väga detailselt muuta.

Blenderi vana videomonteerijat, mis töötab kõrgemal tasemel, st tervete valmis filmilõikudega ehk ribadega korraga, kasutakse video loomise hilisemas faasis. Blenderi erinevad osad töötavad tavapärase tööprotsessis omavahel koos:

- Modelleeri objektid
- Määra materjalid ning lisa valgus, et neile värvi anda
- Animeeri oma objektid, et neid liigutada
- Renderda kaamerate abil video kihid
- Kasuta komposiitimisõlmi:
 - Et pilte värvide kohandamise ja eriefektide abil parandada
 - Ühendada pildid komposiititud pildijadaks (riba)
- Kasuta videomonteerijat, et need videoribad kokku monteerida ja filmiks muuta.

Blenderi videomonteerija on täielik video monteerimissüsteem, millega saad kombineerida mitmeid videokanaleid ning neile efekte lisada. See on olnud Blenderis olemas programmi algusaegadest saadik. Kuigi selle võimaluste arv on piiratud, saad sellega luua muljetavaldavaid montaaže (eriti kui kombineerid seda Blenderi võimsa animeerimissüsteemiga!). Lisaks on võimalik seda pluginate süsteemi abil laiendada ning seeläbi kasutada teda lugematu arvu erinevate pildimanipulatsioonide tegemiseks.

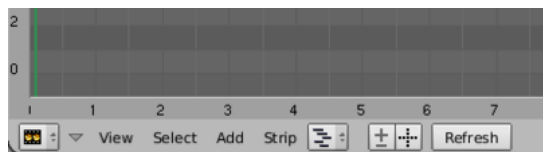
Videomonteerija kasutamisel laed sa sisse erinevad videoklipid ja asetad nad üksteise järele (või mõnelele puhkudel üksteise peale) ning lisad hajumisi ja üleminekuid selleks, et ühe klipi lõpp teise klipi algusega ühendada. Kõige lõpuks saad lisada heliriba, mille abil võid oma videomontaaži ajastused heliga sünkroniseerida. Videomontaažihalduri lõpp-produktiks on sinu valmis film.

FFMPEG tugi

FFMPEG abil avi/quicktime-formaadis filmi eksportimine töötab praegu (versioonis 2.44) ainult Linuxis ja Windowsis. FFMPEG toe puhul saad salvestada oma heliriba koos videoga.

Monteerija ülevaade

Blenderi videomonteerija on paindlik töölaud videoklippide monteerimiseks. Seda kasutatakse klippide ülevaatamiseks ning paljude filmiklippide omavahel kokku kleepimiseks. Temaga on kaasas mitmed sisseehitatud töövahendid (ja pluginatega saab neid piiramatult juurde lisada), mille abil saab teha üleminekuid ühest klipist teise ning mis pakuvad hollywoodilikke professionaalse väljanägemisega efekte.



Videomonteerija montaažirežiimis

Videomontaažihalduril on päis (kus näidatakse menüüd ja vaaterežiime) ja tööala, kus on võimalik kasutada eri vaaterežiime. Menüü Marker (tähis) abil saab monteerijasse lisada tähiseid. Kõigil animatsiooniiredaktoritel on ühised tähised. Loe lähemalt [tähiste peatükist](#).

Monteerimise tööala on jagatud horisontaalselt kanaliteks ja iga videoriba asetatakse ühte neist kanalitest. Iga kanali vasakus servas on tema number. Numbrid algavad nullist (sellesse erilisse kanalisse ei saa sa midagi ise panna!) ning jätkavad kasvavas järjekorras. Paremal oleval pildil on näidatud kolm kanalit (nullist kaheni). Allpool asuvad ribad domineerivad (me kohe räägime sellest). X-suunal näidatakse animatsiooni aega sekundites või kaadrites (ühiku valimiseks vajuta kiirklahvi CtrlT) ja neid kasutatakse aja mõõtmiseks (pildil on näidatud sekundid 1 kuni 7). Aega saad suurendamishuppude või hiireliigutuste abil pikendada või lühendada (loe täpsemalt sissejuhatavatest peatükkidest).



Lülita montaaživäljund sisse

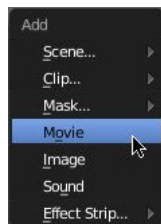
Kui vajutad pildi või video renderdamise nupul, laseb Blender valida, millist pilti sa tahad parajasti valitud kaadrist/kaadritest kokku panna:

- Parajasti valitud stseeni kihi tulemus
- Monteerija kanali 0 tulemus
- Komposiitmissõlmede redaktori renderduskihi tulemus

Videomonteerija on vaikimisi sisselülitatud.

Monteerija kasutamine

Ribade lisamine



Lisamise menüü

Peamine menüü videomonteerijasse sisu lisamiseks on *Add* (lisa). Üldiselt laed sa oma ribad, lood üleminekuefektide jaoks lisaribad ning animeerid seejärel oma montaaži, valides *Do Sequence* (monteeri klipp) ja vajutades nuppu *Anim* (animeerimine). Võid kasutada päises olevat lisamise menüüd (*Add*) või liigutada oma hiirekursori montaažiakna tööala kohale ning vajutada kiirklahvi ⇧ ShiftA.

Klipid võivad olla hiigelsuured

Kolme minuti pikkune quicktime'i .mov-fail võib olla 140 megabaiti. Selle laadimine isegi üle kiire kohaliku võrgu võib aega võtta. Ära arva, et su arvuti või Blender on kinni külmunud, kui tükk aega mitte midagi ei toimu.

Kõigepealt lisame klipi:


- Videoklipp AVI ehk "Audio-Video Interleaved" formaadis (*.avi fail)
- Videoklipp Apple'i QuickTime'i formaadis (*.mov)
- Üksik seisev pilt, mida korratakse mitme kaadri jooksul (*.jpg, *.png jne)
- Nummerdatud piltide jada (*.0001.jpg, *.0002.jpg, *.0003.jpg vms ükskõik millises pildiformaadis)
- Mõnes kataloogis asuv pilt või pildid
- Sinu .blend-failis olev stseen.


Blenderil ei ole vahet, millist neist sa kasutad; võid neid vabalt omavahel segada. Videomonteerijas saavad kõigist neist värvikoodiga ribad:

- Sinist kasutatakse Avi/mov-koodekiga ribade puhul
- Hall tähistab üksikut pilti, mida korraldatakse/kopeeritakse
- Lilla on piltide jada või piltide grupp, mida mängitakse üksteise järel
- Roheline on audio rada

Kui valid ühe neist, muutub videomonteerija aken failisirvi jaoks, mille abil saad valida soovitud fail(d). Toetatud formaadiga failide nimede kõrval on väike ruuduke (sinine piltide, roheline klippide puhul), mis toimib visuaalse vihjena, et nende valimine on võimalik:

Videote või piltide lisamine

Et valida video või video koos heliga, klõpsa selle peal LMB , nii et faili nimi ilmub ülaserivas olevale tekstiväljale - nii valid sa **üksiku** faili (näiteks ühe video)




(Numereeritud) piltide **jada** puhul võid sa: **Valida kataloogi**: klõpsa RMB  kataloogi nimel ning sisse loetakse kõik kataloogis olevad failid vastavalt kataloogi sorteerimise järjekorrale, nii et iga pilt vastab ühele kaadri. **Valida mitu faili järjest**: mine mõnda kataloogi ning lohistage paremat hiirenuppu all hoides üle failinimed, et neid selekteerida. Võid liikuda kataloogis edasi ning jätkata lohistamist, et faile lisaks valida. **Valida mitu faili suvaliselt**: kui klõpsad Shift-klahvi all hoides parema hiireklahviga kataloogis olevatele failidele, siis loetakse nad vastavalt kataloogi sorteerimise järjekorrale sisse, nii et iga pilt on üks kaader. Võid lisada erinevat tüüpi faile (jpg, png, exr, jne). **Valida kõik**: vajuta kiirklahvi A, et valida kõik kataloogis olevad failid või see valik tühistada.

Vajutades nupule Select <misiganes>, muutub aken uuesti videomonteerijaks, kus uus loodav riba on sinu hiire kursori külge kleebitud. Mitut video ei saa korraga parema hiireklahviga valida; kui niimoodi valid, ei loeta neist ühtegi. Parema klahviga vajutamine töötab ainult piltide puhul.

Error: The selected file is not a movie or FFMPEG support not compiled in!

See veateade (Viga: Valitud fail ei ole video või selle jaoks ei ole kompileeritud FFMPEG tuge!) tähendab, et fail on video, mida Blender ära ei tunne või **sa valisid selle vale nupuga**. Saad selle veateate, sest vajutasid videofaili peal *paremat* klahvi VÕI sul puudub koodek, millega valitud avi-faili dekodeerida. Kui tegemist on viimasega, otsi koodek, millega saad faili väljaspool Blenderit maha mängida ning seejärel saad seda ka laadida. Kui tegemist on esimese veaga, pead videod valimiseks vajutama vasakut klahvi.

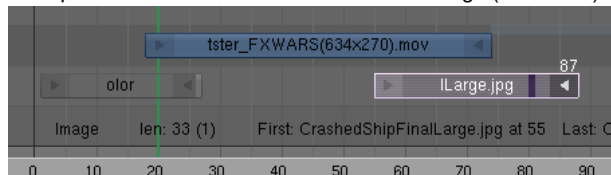
Videomonteerijasse failide lisamiseks vajuta videote peal vasakut klahvi, üksikute piltide peal vasakut klahvi või kasuta pildijadade puhul paremat klahvi ja lohistamist. Liiguta oma hiir õigesse ritta ja soovitud ajale/kohale ning vajuta hiireklahvi, et riba lahti lasta ja kohale asetada (selles kanalis ja valitud kaadrist alates).

Kui lisad pildi, muudab Blender selle 50 kaadri pikkuseks ribaks, mis tähendab, et pilt kestab su videos kaks sekundit (25 fps PAL-video puhul). Lisaks ümberpaigutamisele soovid sa seda sageli ka pikendada või lühendada - seda saad sa teha, klõpsates alguse või lõpu noolel RMB  ja lohistades hiirt paremale või vasakule. Hiirt liigutades värskendub ka kaadri number ning näitab sulle, kus nool parajasti asub. Klõpsa LMB , et liigutamine kinnitada, või RMB , et muudatus tühistada.



Erinevate suurustega hakkama saamine

Erineva suurusega piltide ning erineva suurusega väljundite kasutamine võib olla keeruline. Mõtle nagu piksel. Kui sinu sisendpildi ja renderduse väljundi suurus on erinevad, proovib videomonteerija pildi mõõtkava automaatselt muuta, nii et pilt mahuks täielikult väljundisse. Selle tulemusena võidakse pildi osad ära lõigata. Kui sa seda ei soovi, kasuta sisendipaneeli (Input) olevaid lõikamise (Crop) ja/või nihutamise (Offset) valikuid, et pilti liigutada ning valida sellest väljundi jaoks sobiv ala. Kui kasutad lõikamist (Crop) või nihutamist (Translate), lülitatakse automaatne mõõtkava muutmine välja ja sa saad riba pildisuurust muuta käsitsi teisendamise efektiga (Transform).



Kui sa kerid tööala üles, näed informatsioonikanalit (kanal 0 vertikaalses asukohas), mis sisaldab mõningaid abistavaid vihjeid aktiivse riba kohta. Ülaltoodud näidisel on kaadrites 1 kuni 25 värviriba, seejärel mov-fail ja siis pildiriba. Informatsioonikanal näitab kasulikku informatsiooni pildiriba kohta, mille nimi on riba sees kokku surutud, kuid on informatsiooni ribas täies pikkuses välja toodud.

9999 kaadrit läheb mööda (ainult pildiribade puhul!)

OK, see oli nüüd üks väga ähmane viide laulule, mis räägib 99 õhupallist, kuid me ei ole tõepoolest seda ette näinud, kui kiiresti on Blenderist saanud igapäevane videoredaktor. Kahjuks piirasime me alguses iga videopildi jada failinime ainult nelja numbrikohaga. Kuigi seeläbi on võimalik kasutada kuni 400 sekundit videot (USA formaadis on see umbes 5 minutit), siis Blenderi kasutamiseks filmide puhul pead sa jagama pildiribad nelja numbrikohalisteks lõikuteks ning lisama ise viienda numbrikoha (10 000-19 999), (20 000-29 999) jne. Pane tähele: see puudutab hetkel ainult pildiribasid. Kõik teised riba tüübid töötavad korralikult kuni 300 000 kaadriga (umbes 3 tundi, loe: Ben Hur :).

Koodekid

Sul peab arvutis olema koodek, mis suudab vajalikke avi-faile dekodeerida. Blender ei halda neid. Võid näiteks laadida aadressilt www.xvid.org endale XviD-i koodeki.

FFMPEG tugi

Kui kasutad Blenderit, millel on FFMPEG tugi, saad laadida audio- ja videoribasid koos; valid Movie+Audio(HD) (film + heli) ja kui lisad vastava riba monteerijasse, jagatakse see eraldi audio ja video kanali ribadeks.

Stseeni lisamine

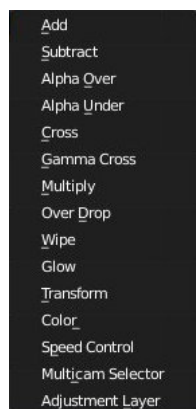
Võid lisada ka oma avatud .blend-faili stseeni virtuaalse pildi väljundi. Vali rippmenüüst vajalik stseen ning selle riba kleebitakse su hiire külge samuti nagu video või pilt. Riba pikkuse määravad selle stseeni animatsiooni sätted (mitte parajasti avatud stseeni omad, kui videomonteerija ei tööta just samas stseenis).

Kui lisad stseeniriba, pea meeles, et selleks, et Blender saaks sulle seda riba videomonteerijas pildi eelvaate režiimis näidata, peab ta stseeni ära renderdama. Keerulise stseeni puhul võib selleks kuluda aega ning seetõttu võib stseeni valimise ning riba tekkimise vahel olla viivitus. Viivituse vähendamiseks lihtsusta stseeni renderdamist valides renderdamiseks vähem kihte.

Heli lisamine



Videomonteerijas saab lisada helikanali, mida sa video asukohta muutes kuuled. Lisa heliriba, kui soovid oma video/animatsiooni vastavalt sellele ajastada (või vastupidi). Lisainfo saamiseks loe [helimontaaži peatüki](#).

Efektide lisamine



Kättesaadavad
sisseehitatud efektid

Blender võimaldab kasutada kahte tüüpi efekte: sisseehtatud ja plugin-efektid. Sisseehtatud efektid on loetletud paremal. Need on Blenderisse sisse ehitatud ning igaüks saab neid kasutada. Plugin-efektid on eraldi failid, mis asuvad su arvuti montaažipluginate kataloogis ning mis laetakse vastavalt vajadusele. Kuigi Blenderi installimisel lisatakse tavapärane pluginate kollektsioon, võib kasutatav kollektsiooni olla eri arvutites erinev.

Iga sisseehtatud efekt seletatakse järgmisel lehel eraldi lahti, kuid nende lisamine ja muutmine toimub alati ühtviisi. Efektiriba lisamiseks vali mõni baasriba (pilt, video või stseen) sellel RMB  klõpsates. Mõnede efektide puhul, nagu üleminek (Cross), pead sa valida kombinatsiooniga ⇧ Shift RMB  teise kattuva riba (see sõltub efektist, mida soovid kasutada). Seejärel vali Add -> Effect (lisa->efekt) ning vali ilmuvast rippmenüüst efekt, mida soovid. Kui see on tehtud, näidatakse efektiriba lähteribade kohal. Kui see on iseseisev efekt, nt värvigeneraator (seda seletatakse hiljem), kleebitakse riba su hiire külge ning selle asetamiseks pead vajutama hiirenuppu.

Kuna enamus efektiribasid sõltuvad ühest või kahest baasribast, sõltub nende kaadriplane asukoht ja kestus aluseks olevatest klippidest. Seega ei pruugi sa saada neid liigutada; pead efektiriba muutmiseks liigutama baasribasid.

Et kasutada efekte, mis kombineerivad või tekitavad kahe aluskihi vahel üleminekuid (või komposiite), pead sa valida nad kas piirdkastiga (B) või kombinatsiooniga shift ja parem hiireklõps. Kui sa efektiriba lisad, asetatakse see nende kohal olevasse kanalisse liigutamise režiimis (vajuta tema paigaleasetamiseks hiirenuppu). Tema maksimaalne kestus on kahe riba kattumise pikkus.


Mõnede efektide (nt AlphaOver) puhul on ribade valimise järjekord oluline. Samuti saad kasutada ühte efektiriba teise sisendina või baasribana ning seeläbi asetada efekte kihiti üksteise peale.

Märkus: ainsaks erandiks on värvigeneraatori efekt. See ei sõltu baasribast - saad seda lisada ja paigutada teistest ribadest sõltumatult. Ka selle pikkust saad sa muuta nagu kõigi harilike ribade oma.

Mode: Montaaž, efektiriba on valitud

Hotkey: C

Menu: Strip -> Change Effect (riba -> muuda efekti)

Kui sa valisid menüüst vale efekti, saad seda alati muuta, selekteerides riba (RMB ) ja valides siis menüüst Strip->Change Effect (riba->vaheta efekt). Või võid vajutada klahvi C, et valitud efektiriba efekti vahetada.

Plugin-efektide lisamine

animglow.dll
boxblur.dll
chromakey_rgb.dll
chromakey_yuv.dll
desaturate.dll
diff.dll
displacer.dll
fadetobw.dll
gimpit.dll
glow.dll
invert.dll
iris.dll
lsc.dll
pmotion.dll
readvertex.dll
rgstereo.dll
robocop.dll
scatter.dll
showzbuf.dll
split.dll
strobe.dll
sweep.dll
videopatterns.dll
wipeout.dll
zblur.dll

Montaažipluginad on spetsiaalsed väikesed koodijupid, mille on kirjutanud erilised programmeerijad keeles C ja mis kujutavad endast dünaamiliselt laetavaid teeke (dynamic load library, DLL). DLL-i saab laadida igal ajal (dünaamiliselt) vastavalt vajadusele ning see *lisatakse* Blenderile. (Kui sind see huvitab, siis sõltub nende laiend sinu platvormist. Linuxis nimetatakse neid näiteks .so (shared object ehk jagatud objekt).)

Paremal olev pilt näitab montaaži DLL-faile, mis on minu süsteemis olemas. Igaüks neist teostab mõnda eriefekti, millele viitab nende nimetus või mis on lahti seletatud [Blenderi pluginate leheküljel](#) või nende programmeerija kodulehel. Näiteks iirise plugin tekitab kahe riba vahele ülemineku, avades esimese keskele suureneva ringi, millest teine riba läbi paistab - nagu kaamera avanev iiris. Mõned neist pluginatest võivad olla viis või enam aastat vanad ja töötavad ikka veel väga hästi. Blender proovib olla tagasiühilduv ning seetõttu peaks nad töötama väljundi formaadist või lahutusest (suurus) sõltumatult.

Riba omadused

Riba omadusi saab näha ja muuta omaduste paneelis, kiirklahv N.

- Muuda (*Edit*) - Muuda riba omadusi
- Sisend (*Input*) - Kust pilte sisse võtta
- Efekt (*Effect*) - Efektiribade sätted
- Filter - Pildi eeltöötlus
- Asendaja (*Proxy*) - Aeglasemate arvutite jaoks mõeldud lahendus, mis näitab päris piltide asendajaid
- Stseen (*Scene*) - Sätted, mis ilmuvad siis, kui valitud on stseeni sisaldav riba
- Heli (*Sound*) - Heliklipi sätted

Kõigi nende valikute paneele ning neis olevaid seadeid näidatakse paremal.

Riba muutmise paneel (Edit Strip)

Name (nimi)

Selle abil saad oma ribasid nimetada või ümber nimetada.

Type (tüüp)

Näitab valitud riba tüüpi.

Blend Mode (segamisrežiim)

Vaikimisi riba asendab (Replace) madalamal tasemel olevate ribade väljundpildi. Kuid vastavalt riba tüübile saab kasutada paljusid teisi segamisrežiime. Näiteks Alpha-Over asetab pildi automaatselt kihina madalama taseme ribade peale. Need automaatsed segamisrežiimid muudavad eraldi efektiribad tarbetuks. Segamise protsent määrab selle, kui palju riba (ka ajaliselt) väljundit mõjutab.

Opacity (katvus)

Määra riba katvus/läbipaistvus.

Mute (lülita välja)

Peidab riba, nii et see ei osale lõplikus pildi arvutamises

Lock (lukusta)

Ei lase riba liigutada.

Channel (kanal)

Muudab riba kanali numbrit ehk rida.

Start Frame (alguskaader)

Muudab riba alguskaadri numbrit ehk teeb sama, mida riba haaramine ja selle liigutamine. Näpunäide: riba lisamisel meeldib mulle see lihtsalt kuhugi asetada ning kasutada seejärel seda välja, et asetada riba soovitud kaadrisse, selle asemel et proovida seda täpselt õigesse kohta lohistada.

Length (pikkus)

Määrab, kui palju kaadreid riba kestab.

Kasuta nuppu Convert to Premul (konverteeri eelkorratuks), kui ribal on olemas alfa (katvuse) kanal. Kasuta seadet FilterY (Y-filter), kui riba pärineb videost ning sellel on paaris või paaritu ülerealaotus (interlaced fields). Suurenda värvide küllastust välja Mul (korruta) abil. Mängi riba tagurpidi, lülitades sisse tagurpidi kaadrite valiku (Reverse Frames). Lase Blenderil näidata iga n. kaadrit, sisestades strobo väärtuse (Strobe). Lõpuks, kui riba pärineb MPEG videost (VCD, DVD, XVid, DivX, ...), ehitatakse tegelik pilt valmis mõne kaadri jooksul - kasuta eelotsimise välja (Preseek), et lasta Blenderit tahapoole vaadata ning koostada pilt eelmise n kaadri põhjal (näiteks **15** Mpeg2 DVD puhul).

Efektiriba (Effect Strip)

Kõigi efektide puhul kasuta riba omaduste paneeli efektiriba muutmiseks. Igal efektil on erinevad seaded, kuid neid kõiki saab muuta omaduste paneelist. Muuda riba pikkust, et muuta teisenduse toimumise kiirust. Hoolimata sellest, kas efekti ribad on sisse ehitatud või plugin-efektid, teevad nad kõik teatavat sorti pildimanipulatsioone, töötades enamasti mõnes teises kanalis oleva klipi või klippide baasil. Efektiriba kuvatakse mõnes kanalis, kuid efekti tulemus on nähtav kanalis 0.

Riba sisend (Strip Input)

Määrab riba allika. Selle väljadeks on faili asukoht, faili nimi, pildi nihe ja lõikamise sätteid.

Siitkaudu saad muuta/värskendada riba poolt kasutatava faili asukohta. See on väga kasulik, kui muutsid kas alusfaili või .blend-faili asukohta - nii ei pea sa riba kustutama ja uuesti looma!

Sul on asukoha jaoks kaks välja - esimene neist tähistab selle kataloogi asukohta, kus fail asub (Path) ning teine on faili enda nimi.

Filter

Selle kaudu saad kiiresti määrata tavapäraseid pildi eeltöötluse sätteid.

Strobe (stroboskoop)

Flip (peegelda)

X peegeldab (keerab teisipidi) pildi vasakult paremale, Y ülalt alla.

Backwards (tagurpidi)

Pöörab riba pildijada tagurpidi

De-Interlace (ülerealaotuse eemaldamine)

Kombineerib videofaili poolkaadrid.

Saturation (küllastatus)

Suurendab või vähendab pildi küllastust.

Multiply (korruta)

Korrutab värve selle arvuga.

Premultiply (eelkorruta)

Eelkorrutab värvikanalid alfakanaliga.

Convert Float (konverteeri ujukomaarvudeks)

Konverteerib sisendi ujukomaarvudeks.

Use Color Balance (kasuta värvitasakaalu)

Selle kaudu pääsed ligi kolmele filtrile, mille abil saad muuta värve: tõste (Lift), gamma (Gamma) ja võimendus (Gain). Igal käigul võib olla positiivne või tagurpidine efekt, kui kasutad vastavat nuppu (Inverse). Määra efekti tugevus, muutes värvinäidist; valge (RGB 1,1,1) tulemust ei mõjuta.

Asendusriba omaduste paneel (Proxy Strip Properties)

Asenduspilt on väiksem pilt (see laetakse kiiremini), mis tähistab peamist pilti. Kasutades asenduste taasehitamist (Rebuild proxy), arvutab Blender suurte piltide jaoks välja väikesed pildid (nagu eelvaated) ning selleks võib kuluda natuke aega. Kuid pärast nende arvutamist on montaaživõtted (nt kaadri vahetus ja kerimine) ning komposiitmisvõtted (nt üleminek), mis kasutavad neid asenduspilte, palju kiiremad, kuid tulemus on madala lahutusega. Lülita enne lõplikku renderdamist asendusribad välja.

Heli (Sound)

See paneel ilmub siis, kui valitud on helifail.

Siit saad määrata heliriba faili asukoha ja nime.

Pack (paki)

Pakib helifaili parajasti avatud .blend-faili.

Caching (puhverdamine)

Helifail dekodeeritakse ja laetakse operatiivmällu.

Volume (helitugevus)

Määra helifaili helitugevus.

Attenuation/dB (vaibumine/dB)

Helitugevuse vähendus detsibellides.

Trim Duration: Start/End (lõika kestus)

algus/lõpp)

Nihutab heliriba algust ja/või lõppu.

Stseen (Scene)

Määra stseen, millele valitud stseeniriba lingib.

Sequencer (monteerija)

Saada renderduse (ja komposiidi) tulemus läbi videomonteerija, kui vastava stseeni ribad eksisteerivad. See on sama funktsioon mis renderdamise seadetes olev.

Camera Override (asenda aktiivne kaamera)

Muuda kasutatavat kaamerat.

Vaate muutmine

Kasuta neid kiirvalikuid, et muuta videomonteerija vaadet.

Liiguta

MMB 

Suurenda/vähenda

Wheel 

Vertikaalne kerimine

{Shortcut|Shift|wheel}} või lohista vasakpoolset kerimisriba.

Horisontaalne kerimine

Ctrl Wheel  või lohista alumist kerimisriba.


Vaate vertikaalse mõõtkava muutmiseks lohista vertikaalses kerimisribas olevaid ringe.

Vaate horisontaalse mõõtkava muutmiseks lohista horisontaalses kerimisribas olevaid ringe.


Nagu tavaliselt, määrab ka videomonteerija menüü View (vaade) selle, mida ja kuidas sa tööalas näed.

Omaduste paneel (Properties Panel)

Omaduste paneelis on seaded, mis määravad selle, kuidas eelvaadet kuvatakse.

View all Sequences,  Home (näita kõiki ribasid)

Vähendab vaadet, et näidata kõiki ribasid.

Fit preview in Window,  Home (mahuta eelvaade aknasse)

Muudab eelvaate suurust, nii et see mahub aknasse.

Show Preview 1:1, 1 NumPad (näita eelvaadet 1

1)

Muudab eelvaate suuruse 1:1-le (tegelik suurus).

View Selected, . NumPad (näita valitud)

Suurendab vaadet, nii et nähtaval oleks ainult valitud ribad.

Kasuta seda võimalust, kui töötad paljude ribadega korraga ning soovid tööks kasutada tervet ekraani.

Mode: Montaaž

Hotkey: T

Menu: View -> Show Frames, View -> Show Seconds (vaade -> näita kaadreid, vaade -> näita seundeid)

Draw Frames (näita kaadreid)

Näitab kaadrinäidikus aja asemel kaadri numbrit.

Show Frame Number Indicator (näita kaadrinäidikut)

Muudab tööala allservas olevat mõõtühikut vaheldumisi sekunditeks ja kaadriteks.

Safe Margin (ohutu serv)

Näitab eelvaate peal kihina serva, mis märgib seda ala, mille sisse võib ohutult panna tiitreid.


Separate Colors (eralda värvid)

Kasutades heleduse lainevormi, eraldab see valik R, G ja B kanalid eraldi graafikuteks.

Transform Markers (teisendustähised)

Näitab koos ribadega ka teisendustähiseid.

Kerimine

Oma filmis edasi ja tagasi liikumiseks kasuta ajatelje akent. Klõpsa LMB  ja lohista hiirt ajateljes vasakule/paremale, et liigutada vertikaalset riba, mis tähistab parajasti valitud kaadrit. Selle peale näidatakse videomonteerija aknas kaadri vastavat pilti.

Reaalajas kerimine ja pildi näitamine on võimalik suhteliselt tavaliste arvutitega, kui vaatad pildijada või video (avi/mov) faili. Stseeni pildid tuleb kõik eraldi renderdada, mis võib võtta mõnda aega.

Vaaterežiimid

Päises olevate ikoonidega saad muuta videomonteerija vaate tüüpi. Vaikimisi näidatakse montaaživaadet. Teine nupp näitab ainult eelvaate akent ning kolmas nii montaaživaadet kui ka eelvaadet.

Kui sisse on lülitatud eelvaade, on sul valikud, millist eelvaate tüüpi näidata. Need on lahti seletatud [vaaterežiimide lehel](#).

Stseeni eelvaade

Kui kasutad montaažis stseeniriba, määravad omaduste paneelis olevad seaded selle, kuidas seda eelvaate aknas näidatakse.

Open GL Preview (Open GL-i eelvaade)

Kui sul on olemas Open GL, lülita see seade sisse, et kasutada Open GL-i stseeni eelvaate renderduste puhul.

Ripppenüüst saad valida, kuidas stseeni näidatakse (piirdkast, traatvõre, varjutatud, tekstuuriga).

Vaate seaded

Omaduste paneelis olevate vaate seadete all on olemas vaate lisavõimalused.

Show Overexposed (näita ülesäritust)

Suurenda see number üheks või enamaks ning eelvaate pildile asetatakse triibuline lisakiht, mis näitab, kus kohas pilt on üle säritatud. Suurem number tõstab ülesärituse näitamise läve.

Safe Margin (ohutu serv)

Näitab eelvaate peal kihina serva, mis märgib seda ala, mille sisse võib ohutult panna tiitreid.

Proxy Render Size (asendusrenderduse suurus)

Joonistab eelvaate, kasutades kas täislahutust või erinevaid asenduslahutusi. Renderduslahutus määratakse renderdamise seadete paneelis. Väiksema eelvaate suuruse kasutamine suurendab kiirust.

Värskenda vaadet

Mõned tegevused (nagu objekti liigutamine 3D-vaates) ei pruugi sundida montaaživaadet enda renderduspilti värskendama (sest

liigutamine ei pruugi renderdatud pilti mõjutada). Kui pilti või videot, mida ribana kasutatakse, mõnes Blenderi välises programmis muudetakse, ei saa sinu operatsioonisüsteem sellest Blenderit teavitada. Et sundida Blenderit neid faile uuesti lugema ning panna teda 3D-vaadet uuesti renderdama, vajuta värskendusnupule (Refresh), mis käsib Blenderil värskendada ja sünkroniseerida kõik puhverdatud pildid ning arvutada uuesti parajasti valitud kaader.

Ribade valimine

Valikumenüü (Select) abil saad sa valida ribasid erinevatel viisidel.

Strips to the Left (vasakul olevad ribad)

Vali kõik parajasti valitud ribast vasakule jäävad ribad.

Strips to the Right (paremal olevad ribad)


Vali kõik parajasti valitud ribast paremale jäävad ribad.

Select Surrounding Handles, AltCtrl RMB  (vali ümbritsevad käepidemed)

Vali mõlemad valitud riba käepidemed ning nende kõrvale jäävate ribad naaberkäepidemed. Selle meetodiga võid selekteerida siis, kui soovid liigutada kahe riba vahele jäävat riba, kuid ei soovi selle pikkust muuta.

Left Handle, Alt RMB  (vasak käepide)

Valib parajasti valitud riba vasaku käepideme.

Right Handle, Ctrl RMB  (parem käepide)

Valib parajasti valitud riba parema käepideme.

Linked (lingitud)

Valib kõik parajasti valitud ribaga lingitud ribad.

Select All, A (vali kõik)

Valib kõik ribad.

Select Inverse (pööra valik tagurpidi)

Pöörab praeguse valiku tagurpidi.


Border Select, B (vali piirdkastiga)

Alustab piirdkasti (Box) režiimis valimist. Vajuta ja lohista hiirt, et tõmmata montaaži tööalasse jäävate ribad ümber nelinurkne lasso. Kui lased hiirenupu lahti, valitakse kõik alasse jäävad lisaribad.

Ribade liigutamine ja muutmine

Kiirklahviga G saad liigutada valitud riba(sid) ajaliselt või kanalite vahel. Liiguta hiirt horisontaalselt (vasak/parem), et muuta riba asukohta ajas. Liiguta vertikaalselt (üles/alla), et vahetada kanalit. Liigutades riba üles (kõrgemale kanalile) ning teise ribaga kohakuti, siis seda kattumise ulatuses ei näidata; madalamate kanalite ribasid näidatakse kõrgemate kanalite ees. **HOIATUS: see on täpselt vastupidine sellele, mis toimub enamikus teistes videoreaktorites!** Selleks et nad mõlemad näha oleks, pead neid segama, kasutades lisamist (Add), korrutamist (Multiply), lahutamist (Subtract) või mõnda muud komposiitmisefekti.

Kui oled lisanud riba ekslikult ja ei soovi seda enam kasutada, kustuda see, vajutades kiirklahvi X või kasutades menüü valikut.

Ribast linkimata koopia tegemiseks duplikeeri (Duplicate) see, lohista sobilikku aega ja kanalisse ning aseta hiirenupu LMB  klõpsuga paigale.

Menüüs Strip (riba) on järgmised ribadega töötamise lisatööriistad:

Grab/Move (haara/liiguta)

Grab/Extend from Frame (haara/pikenda kaadrist edasi)

Cut (hard) at frame (lõika (tugevalt) kaadris)

Cut (soft) at frame (lõika (pehmelt) kaadris)

Separate Images (eralda pildid)

Deinterlace Movies (eemalda ülerealaotus)

Duplicate Strips (duplitseeri ribad)

Erase Strips (kustuta ribad)

Set Render Size (määra renderduse suurus)

Make Meta Strip (loo meta-riba)

UnMeta Strip (tühista meta-riba)

Reload Strips (lae ribad uuesti)

Reassign Inputs (määra uuesti sisendid)

Swap Inputs (vaheta sisendid)

Lock Strips (lukusta ribad)

UnLock Strips (vabasta ribad)

Mute Strips (lülita ribad välja)

Un-Mute Strips (lülita ribad sisse)

Mute Deselected Strips (lülita välja valimata ribad)

Snap Strips (pane ribad nakkuma)

Swap Strips (vaheta ribad)




Kaadriga nakkumine

⇧ ShiftS Aseta kursor (vertikaalne roheline joon) vajaliku ajaga kohta. Pane riba valitud kaadriga nakkuma, et see alustaks täpselt sellest kaadrist. Kui aega näidatakse sekundites, saad näha sekundi murdosi, kui suurendad vaadet: nii võid jõuda kuni üksikute kaadrite tasemeni.



Eralda pildid ribadeks

Y konverteerib riba mitmeks ribaks, kus iga riba vastab ühele kaadriks. See on väga kasulik presentatsioonide ja teiste sarnaste juhtude puhul, kus soovid tuua sisse komplekti omavahel mitte seotud pilte.

Ribade muutmine

- RMB  klõps riba keskel valib **terve** riba; hoides nuppu all (või vajutades haaramise kiirklahvi G) ning liigutades hiirt, lohistatakse riba ringi.
- RMB  klõps riba vasakul noolel valib riba **algus**kaadri; hoides nuppu all (või vajutades haaramise kiirklahvi G) ning liigutades hiirt vasakule/paremale, muudetakse riba sisest alguskaadrit vastavalt sellele, mitme kaadri võrra sa liigutad:
 - Kui sul on 20 pildi pikkune jadariba ning sa lohistad vasakut noolt kümne kaadri võrra paremale, algab riba pildist 11 (pildid 1 kuni 10 jäetakse vahele). Kasuta seda kasutu sissejuhatuse või muu musta materjali äralõikamiseks.
 - Vasaku noole lohistamine vasakule loob sissejuhatuse (koopiad) esimesest kaadrist, mis kestab samapalju kaadreid, kui sa lohistasid. Kasuta seda, kui vajad klipile üleminekuks mõningaid lisakaadreid.
- RMB  klõps riba paremal noolel valib riba **lõpp**kaadri; hoides nuppu all (või vajutades haaramise kiirklahvi G) ning liigutades hiirt, muudetakse riba sisest lõpukaadrit:
 - Parema noole lohistamine vasakule lühendab klippi; kõiki ülejäänuid originaalpilte ignoreeritakse. Kasuta seda üleliigse kiireks mahaloikamiseks.
 - Parema noole lohistamine paremale pikendab klippi. Videote ja pildijadade puhul kasutatakse ära ülejäänud materjal, kuni see otsa lõppeb. Kui klippi pikendatakse üle selle lõpu, teeb Blender koopia viimasest pildist. Kasuta seda valitud klipist edasi üleminekuks.

Mitmikvalik

Võid valida mitu riba (või nende käepidet), klõpsates neile kombinatsiooniga  Shift RMB  kui sa nüüd vajutad G, liigub kõik valitu koos hiirega; see tähendab, et võid näiteks samal ajal liigutada ühte riba, lühendada kahte teist ning pikendada neljandat.


- RIBA PIKENDAMINE. Kui valitud on mitu erinevat pildiriba, viib kiirklahv E sind pikendusrežiimi (Extend). Kõik valitud ribade käepidemed, mis jäävad praegusest kaadrist "hiire poole", muutuvad koos, mis tähendab, et seeläbi saad pikendada ribasid, mis langevad parajasti valitud kaadrisse, nii et kõik teised kohanduvad vastavalt.

Kahe riba üksteise otsa asetamine paneb nad üksteisele järgnema, riba lõikamiseks vajuta kiirklahvi K. K vajutamine lõikab valitud ribad valitud kaadris kaheks. Kasuta lõikamist, et eemaldada üleliigsed sisse- ja väljajuhatused või tarbetud kaadrid (kiirklahv "C" oli juba kasutuses muutmise (Change) jaoks).

Märkus 'lõikamise' kohta

Kui sa riba 'lõikad', ei tee sa samasugust lõiget nagu 'vanasti' tehti reaalse filmi puhul. Tegelikult teed sa ribast koopia: originaali lõpp tõmmatakse lõikekoha suunas 'sisse', nagu ka koopia algus.

Kujuta näiteks ette, et sul on **50** kaadri pikkune riba, millest sa soovid esimesed kümme kaadrit kustutada. Pead minema kaadrisse **11** ja vajutama kiirklahvi K - see lõige 'jagab' riba kahte osasse. Võid nüüd valida esimese väikese osa (kaadrid **1** kuni **10**) ning selle kiirklahviga X kustutada.

Sa võid küll arvata, et oled kaadrid **1** kuni **10** päriselt kustutanud, kuid need on ikka veel olemas, nad on tõmmatud 'sisse' - nagu reaalse filmirulli puhul - kaadri numbriga **11** alla: kustutanud oled sa ühe kahest ribast, mis 'lõikega' loodi. Ning sa võid iga kell oma 'kadunud' kaadrid tagasi saada (klõpsa lihtsalt riba vasakul noolel RMB , haara sellest kiirklahviga G ning lohista seda soovitud kaadrite hulga võrra vasakule. Või paremale, et veelgi rohkem kaadreid 'peita' - see on veel üks võimalus, kuidas riba algusest/lõpust kaadreid eemaldada!).

Selline meetod on peaaegu iga montaažiprogrammi aluseks ning see on väga kasulik!

Tegevus peatub

Kui sa pikendad algust või lõppu reaalsest algusest või lõpust pikemaks, pea meeles, et kopeeritakse ainult viimast pilti, nii et vaatamise puhul näib tegevus selle kaadriga peatuvat. Alusta oma üleminekut (hajumine, ristamine) natuke varem, kus tegevus ikka veel toimub, et tegevuse peatumine ei oleks märgatav (kui sa just ei soovi, et tulemus oleks nagu 80ndate draamasarjad).

Muuda efektriba pikkust, muutes selle aluseks olevate ribade alguse/lõpu kaadrit.

Kopeeri ja aseta

Klippi saad kopeerida ja puhvrist monteerijasse asetada, kasutades kahte päises olevat nuppu.

Metaribad

Metariba on ribade grupp. Vali ribad, mida soovid grupeerida, ning muuda nad kiirklahviga M üheks metaribaks. Metariba ulatub esimese klipi algusest viimase klipi lõpuni ning paigutab kõik kanalid ühte ribasse - samaselt sellega, nagu miksiksid audioprogrammis helid kokku. Nende eraldamine (lahtigrupeerimine) asetab nad tagasi algsetesse asukohtadesse ja kanalitesse.

Montaaži esitusrežiimid

Vaikimisi näitab videomonteerija ainult montaaži järjestajat. Päiseribas on mitu valikut, mille abil saad näha monteeritavat materjali reaajas ja seda erinevatel viisidel.

Teine nupp näitab monteerijas ainult eelvaadet ja kolmas nii monteerijat kui ka eelvaadet

Videomonteerijas saad näha mitmesuguseid valitud kaadri komposiitimise tulemusi:

- Image/Sequence (pilt/pildijada): Värvid (mida ka sina ise näed)
- Chroma (värvilisus): Värvide toonid ja küllastus
- Luma (heledus): Heledus/kontrastsus
- Histogram (tulpdiaagramm): Punase, rohelise ja sinise tasemed

Värvilisuse (Chroma), heleduse (Luma) ja pildi (Image) režiimides ilmub kanali valimise nupp; kanalis 0 on tulemus, mis saadakse, kui ribad on komposiititud kokku nende efektiribadega. Kanal 1 näitab seda, milline näeb välja kanalis 1 oleva riba parajasti valitud kaader (kanal 1 on kõige alumine). Nende režiimide poolt näidatav on kas komposiit (kanal 0) või vastava riba vastav kaader (kanali 1 kuni n).

Nendes vaaterežiimides saad vaadet suurendada ja vähendada hiire ratast kerides.

Pildi eelvaade

Montaažiakna ülemises servas on teine videomonteerija paan, mis näitab pildi eelvaadet. See näitab, milline näeb välja lõpuks salvestatav video. See on peamine töötegemise viis, kui tegeled ribade lisamisega ning nende liigutamise, lõikamise, grupeerimise (muutes nad meta-ribadeks) ja eriefektide lisamisega.

Heleduse lainevorm

Selle kuvamiserežiimi puhul näidatakse valitud kanali heleduse jaotust.

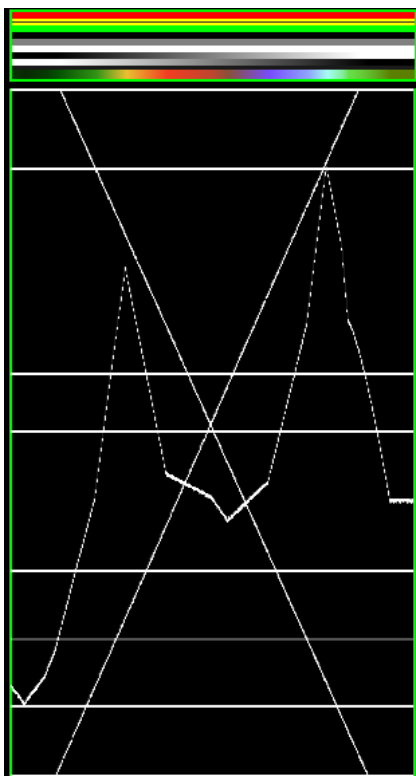
Heleduse lainevormi abil saad veenduda selles, kui kvaliteetselt on heledus üle sinu videosignaali jaotunud. Võid vaadata heleduse lainevormi tavalise väljundrežiimi asemele igas kontroll-monitoris.

See režiim genereerib iga rastrijoone jaoks heleduse graafiku. Need graafikud joonistatakse üksteise peale. Kui jooned ristuvad, muutuvad punktid heledamaks (mis on mitmesaja rastrijoone puhul väga tõenäoline). Mõistad pilti kõige lihtsamalt siis, kui ühendad oma televiisori videoheleduse (Luma) väljundisse ostilloskoobi. Põhimõtteliselt näeb see välja samasugune.

Selles režiimis näitab vertikaalne telg heledust: 0 on all, 1 on üleval; horisontaalne telg vastab kaadri enda horisontaalsele teljele. Kõveraid on sama palju kui kaadris rastrijooni: iga kõver näitab ühe joone pikslite heledust. Lisaks sellele näitab iga piksli värvus selles režiimis ühesuguse heledusega pikslite arvu selles kaadri tulpas - st nende kõverate hulka, mis seda punkti läbivad (must/läbipaistev tähistab, et selliseid piksleid pole, valge/läbipaistmatu, et selliseid piksleid on vähemalt kolm).

See režiim on kasulik:

- Kui lainevorm ei täida tervet pilti, võib olla kasulik liigutada *gamma* pluginas olevaid liugureid "setup" ja "gain", kuni lainevorm täidab terve pildi (automaatne kontrastsuse venitamine).
- Keerulisemate gamma-pluginate puhul saad otsustada, milliste alade küllastust pead vähendama (seda eriti tumedate alade puhul).
- Saad otsustada, kas soovid kogu riba montaažist eemaldada, kuna pilt on moondunud ning ülalt ja alt äärest ära lõigatud.



'Lihtne' pilt.

Heleduse lainevormi (Luma waveform) horisontaalsed jooned vastavad pildi ühes toonis joontele.

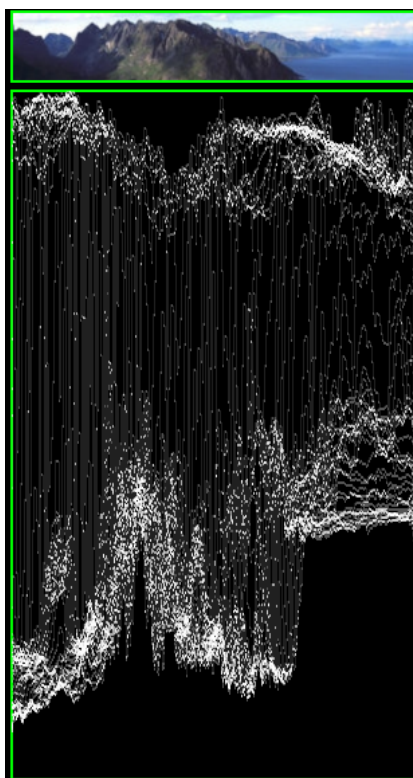
Pane tähele, et ühe piksli laiust '20% halli' joont (kollase riba sees) kujutab heleduse lainevormis hall joon.

Kaks joont, mis moodustavad 'X-i', tähistavad kahte lineaarset värvüleminekut (valge→must ja must→valge).

Lõpuks vastab katkendlik joon esimese pildi allservas olevale keerulisele toonide üleminekule.

Videomonteerija heleduse eelvaadete näited.

Pane tähele, et pildid (üleväl olev esimene roheline kast) on ainult 50 pikslit kõrged, et piirata heleduse lainevormis näidatavate kõverate hulka!



'Päris' pilt.

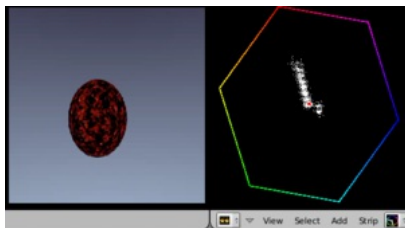
Kõverad on päris hästi nähtavad.

Leidsime, et taeva heledus on 80-100%, mere heledus umbes 40%

ning mägede heledus 10-20%, mis kasvab päikeselises osas umbes 40%-ni.

Kasuta seda režiimi kontrollimaks, kas kõigi kanali kaadrite kontrastsus ja heledus on ühtlane. Kui kohad, kus filmis peaks olema ühtlane valgus, ei ole ühtlased, jääb mulje nagu oleks välgatanud fotoaparaadi välk või oleks sisse lülitatud lisavalgus. See võib juhtuda, kui kaks riba, mis peaksid tegelikult olema samased, renderdati või filmiti erinevates valgustingimustes.

Värvilisuse vektroskoop



Videomonteerija värvilisuse režiimi näidis

Kasuta seda režiimi, et hinnata värvide jagunemise ja küllastuse kvaliteeti. Samuti saad selles režiimis vaadata U/V hajuvusdiagrammi.

Pilt konverteeritakse YUV-formaati. U ja V väärtused tähistavad värvuse nurka. Iga pildi piksli puhul arvutatakse välja nende U ja V väärtuse asukoht diagrammil. Kui mitmel piksilil on sama U/V väärtus, muutub vastav diagrammi piksel heledamaks.

Aitامaks sind mõista, millist värvi mõeldakse, on servadesse joonistatud kuusnurk, mis näitab äärmuslike positsioone (punane, magenta, sinine, tsüaan, roheline, kollane), ja punane rist, mis tähistab nullpunkti (värvuse puudumist).

Teiste sõnadega esitatakse selles režiimis kuusnurga sees pildi valitud kanali värviruumi. Iga kuusnurga punkt on põhivärv: punane, magenta, sinine, tsüaan, roheline, kollane. Must/hall/valge asuvad keskpunktis ning üldise küllastuse suurenemist tähistavad äärele lähenevad punktid. Paremal oleval näidisel sisaldab pilt palju punast (50% küllastust) ja natuke sinist ning rohelist ei ole üldse.

Pea alati meele, et tuleb käivitada lisavaade, mis näitab lõpptulemust. Värv kalibreerimine on maitse asi ja sõltub sinu soovidest.

Kasuta seda režiimi kontrollimaks, kas värvide küllastus pole üliiia suur. Kuigi üleküllastatud pildid näevad arvuti ekraanil või okunstina lähedad välja, paistavad nad suure ekraaniga televisoris jubedad. Kasuta video kerimise jaoks animatsiooni kiirklahvi Alt-A; see režiim uuendab ennast iga kaadri puhul uue või täiendatud kaardistusega. Samamoodi nagu sa kasutada pildi režiimi nägemaks,

milline pilt välja näeb, kasuta värvilisuse vektroskoopi nägemaks, milliseid värve kasutatakse.

See režiim on kasulik:

- Kui su pilt on väga tundeküllane või pastelne, võid kasutada U/V diagrammi. Tõenäoliselt näed, et kõik pikslid koonduvad ühte kohta. Kui kasutad "gamma" pluginat, saad U/V diagrammil näha, kas sa moonutad seeläbi värve.
- Kui teed värvide sobitamist käsitsi, saad selle abil sobitada erinevate kanalite monitorides nähtavat.

Tulpdiagramm

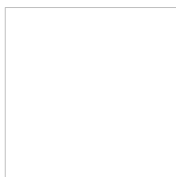
See režiim näitab graafikut, mis kujutab parajasti näidatava pildi pikslite värviinfo jagunemist. X-telg kujutab pikslite väärtusi (nullist üheni või nullist 255-ni) ning Y-telg näitab, mitu pikslit vastab sellele toonivahemikule. Põhiliselt pimedal pildil puhul oleks enamik informatsiooni graafiku vasakus servas.

Kasuta seda režiimi pildi tooniulatuse tasakaalustamiseks. Hästi tasakaalustatud pildil peaks olema sujuv värvide väärtuste jagunemine.

Montaažiefektid

Blenderiga tuleb kaasa 16 sisseehitatud ja seega universaalset efekti. Mõned töötavad kahel ribal, mõned ühel ja mõned loovad uue riba. Iga efekt parandab mingil moel su materjali või võimaldab professionaalsel tasemel üleminekuid.

Liida (*Add*)



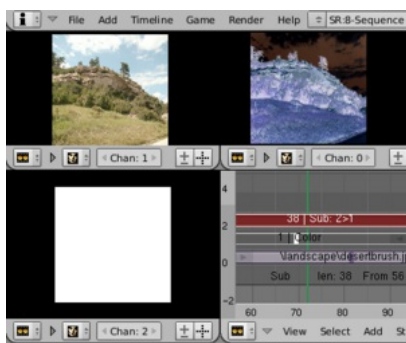
Kas sa
müristamist
kuuled?

Liitmiseefekt Add liidab kokku kaks värvi. Punane ja tsüaan (roheline ja sinine) tekitavad valge. Punane ja sinine tekitavad "magenta" (st lilla!). Punane ja roheline tekitavad kollase.

Liitmiseefekt liidab kahe riba värvid kokku. Kasuta seda aluseks oleva pildiriba ja muutjaribaga. Muutjariba on kas puhas toon, mustvalge mask või mõni täiesti teine pilt. Paremal olev näidis näitab, mis juhtub siis, kui lisad pildile halli ning animeerid efekti ajas. Pilt muutub heledaks, sest me lisasime halli tooni (R:.5, G:.5, B:.5) - näiteks sinisele (R:1, G:.1, B:.5), mille tulemuseks oli (R:.6, G:.6, B:1.0), mis säilitab algse tooni (värvide vahelise suhte), kuid on heledam. Kui rakendada seda efekti tervele pildile, siis näib, nagu pilt välgataks tervenisti.

Võid kasutada seda terve pildi heleduse suurendamiseks või mustvalge maski abil mõnede kindlate pildipiirkondade heleduse suurendamiseks. Segamissõlm (*Mix*) liitmisrežiimis (*Add*) teeb täpselt sama, mida antud juhul liitmise efektriba, ja tema tulemust saab muuta täpselt samamoodi kordaja (*Factor*) muutmise teel.

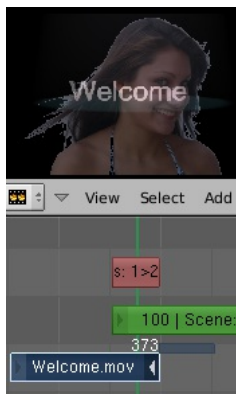
Lahutamine (*Subtract*)



Lahutamine (*Subtract*)

See efekt lahutab ühe riba värvikanalite väärtused teise omadest maha. Sellega saab muuta pildi negatiiviks või vaheta ribade järjekorda ning muuta peamine riba lihtsalt tumedamaks. Sinise tooni lahutamine valgest pildist muudab selle kollaseks, sest punane ja roheline annavad kokku kollase.

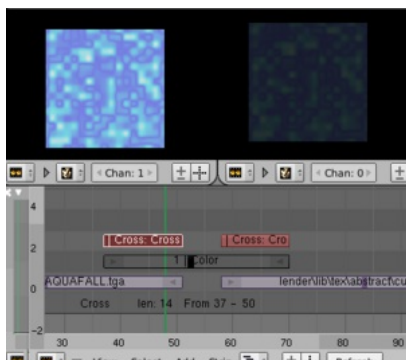
Üleminek (*Cross*) ja gammapõhine üleminek (*Gamma Cross*)



See efekt tekitab sujuva ülemineku ühest ribast teise vastavalt sellele, mitu kaadrit ribadel üksteisega kattuvad. See on väga kasulik efekt, mis sulandab terve pildi ühest ribast teiseks.

Gammapõhine üleminek (*Gamma Cross*) kasutab värvikorrektuuri, mille tulemuseks on silmale meeldivam sujuv üleminek.

Mustaks sulandumine (*Fade to Black*)

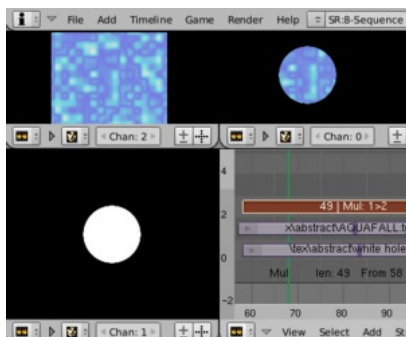


Sulandumine mustaks ja tagasi

Paljud stseenid sulanduvad mustaks ning seejärel uude stseeni, selle asemel et otse ühest teise liikuda.

Paremal on näidatud selle saavutamiseks vajalikku ribade asetust. Kaks videoriba asuvad kanal 1, sa lisad teise kanalisse värvigeneraatori riba (Add->Color Generator - lisa->värvigeneraator), mis kattub kahe peamise riba ühenduskohaga. Muuda värv mustaks ning lisa kaks üleminekuefetti (Cross) - esimene kanalist 1 kanaliks 2 (must) ja teine vastupidi (kanal 2 kanaliks 1). Esimene riba sulandab stseeni mustaks ning seejärel sulandab teine riba pildi mustast uude stseeni. Loomulikult võid üleminekuks kasutada ükskõik millist värvi. Must on rahustav vahepaus; punane tekitab pinget. Kasuta teise riba sisse juhatamiseks tema peamist värvi.

Korrumine (*Multiply*)



Korrumiseefekt (Multiply).

Korrumiseefekt (Multiply) korrutab kahe värvi kanalite väärtused. Blender kasutab värvikanalite jaoks väärtusi vahemikus **0.0** kuni **1.0** ning ei pea selle operatsiooni tulemust normaliseerima, sest kahe vahemikus **0.0** kuni **1.0** oleva väärtuse korrutamise tulemus jääb alati vahemikku **0.0** kuni **1.0** ('tavalise' kolme baidiga esitamise puhul – nagu RGB (**124, 255, 56**) – on korrutamise tulemused liiga kõrged, näiteks RGB (**7316, 46410, 1848**) ning nad tuleb 'tagasi alandada' ehk normaliseerida neid lihtsalt **256**-ga jagades! – et tulemused 'liiguks tagasi' vahemikku **0** kuni **255**...).

Sellel efektil on kaks peamist kasutusvõimalust:

Maskiga

Mask on mustvalge pilt, mis pärast 'tavalise' pildiga korrumist näitab ainult maski valgetele aladele jäävat (kõik muu on must). James Bondi filmide alguskaadrid, kus kaamera vaatab läbi püstolitoru Jamesi suunas, on taolise efekti heaks näiteks.

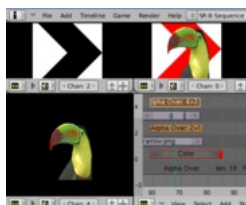
Ühtlaste värvidega

Värvitooni korrumine 'tavalise' pildiga võimaldab sul pildi mõningaid toone pehmedada (ja seega - sümmeetriliselt - teisi tugevdada). Kui sul on näiteks pruun piksel värviga RGB (**0.50, 0.29, 0.05**) ja sa korrutad seda tsüaani filtriga (ühtlane värv RGB (**0.0, 1.0, 1.0**)), saad tulemuseks värvi RGB (**0.0, 0.29, 0.5**). Visuaalselt n-ö tapad sa tulemuses punased ning tõstad esile (läbi 'sümmeetria' – nende tegelikud väärtused jäävad muutmata!) sinised ja rohelised. Füüsikaliselt on efekt sama, mis tsüaanvärvilise valgusti suunamisel šokolaaditahvile. Emotsionaalselt muutub taimestik lopsakamaks, vesi Kariibi mere sarnaseks ja kutsuvaks ning taevast sõbralikumaks.

Märkus

See efekt vähendab pildi üldist heledust (korrumis on alati sama või väiksem kui väiksem tegur). Kui üks piltidest on täiesti valge, on tulemuseks teine pilt; kui üks piltidest on täiesti must, on tulemus täiesti must!

Alpha Over, Alpha Under ja Alpha Over Drop




Alpha Over-efekt

Alfa (katvuse kanalit) kasutades komposiidib see efekt tulemuse, mis sõltub peamise pildi läbipaistvatest aladest. Kui kasutad stseeniriba, siis need pildi alad, kus objekte ei ole, on läbipaistvad - nende alfa väärtus on 0. Kui kasutad filmiriba, on selle alfa

väärtus 1 (täiesti läbipaistmatu).

Seega saad Alpha Over/Alpha Under-efekti abil kolmemõõtmelise stseeni filmi peale asetada. Tulemusena käitub sinu mudel, nagu ta oleks filmi osa. Kordaja (Factor) kõver määrab selle, kui suur osa esiplaanist segatakse taustaga, sulandades esiplaani tausta peale. Läbipaistva esiplaani pildi värve ei arvestata ning nad ei mõjuta tausta värve.

Vali kaks riba (⇧ Shift RMB 

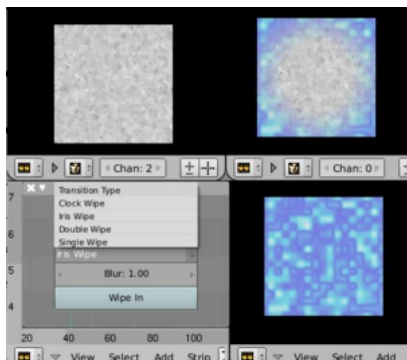
- Kasutades valikut Alpha Over, asetatakse ribad valimise järjekorras üksteise peale; esimesena valitud riba on taustaks ja teine asetatakse esimesena valitu *peale*. Kordaja (Fac) määrab *esiplaani katvuse*, st Fac väärtus **0.0** näitab ainult tausta ning Fac väärtus **1.0** ainult esiplaani (loomulikult välja arvatud esiplaani läbipaistvad osad!)
- Kasutades valikut Alpha Under, toimub asi vastupidiselt - esimesena valitud riba on esiplaan ning teisena valitu tagaplaan. Lisaks määrab kordaja (Fac) *tausta nähtavuse*, st Fac väärtus **0.0** näitab ainult esiplaani (taust on täiesti läbipaistev) ja Fac väärtusega **1.0** annab sama tulemuse nagu Alpha Over.
- Valiku Alpha Over Drop tulemus on kahe teise vahepealne: nagu valiku Alpha Under puhul, on esimesena valitud riba esiplaan, kuid kordaja (Fac) väärtus määrab esiplaani katvuse nagu valiku Alpha Over puhul.

Näidisel on toodud Alpha Over-efekti kihtide asetuse. Kõige alumine kanal on punane ning selle peale on asetatud nool. Need kaks kanalit liidetakse Alpha Over-efekti abil kanaliks 3. Minu lemmiklind tuukan on kanalil 4 ning kanal 5 asetab tuukani Alpha Over-efekti abil komposiititud punase noole peale. Viimane efekt on seotud kanaliga 0, mida renderdatakse.

Vajutades esiplaani riba seadete paneelil nupule PreMult Alpha (alfa eelkorutus), ei korrutata ega liideta kahe riba alfa väärtusi. Kasuta seda efekti, kui lisad esiplaani riba, millel on vahelduv alfakanal (mõned läbipaistmatud alad, mõned läbipaistvad, mõned vahepealsed), riba peale, mille alfakanal on täiesti läbipaistmatu (alfa=1.0 või suurem). Kui näed Alpha Over-efekti kasutades oma esiplaani objektide ümber kuma või esiplaanil imelikke läbipaistvaid alasid, siis lülita eelkorutus (PreMultiply) sisse. Alpha Over Drop-efekt töötab paljuski nagu üleminek, kuid eelistab ülemist ehk teist pilti ning tulemuseks on sujuvam pealeasetamise efekt kui see, mille saavutad ülemineku segamisega. Loomulikult arvestavad kõik alfa efektid ka alfakanalit (katvus), samas kui üleminek seda ei tee.

Katvuse (Alpha) määra ja seega värvide segamist saab määrata F-kõveraga. Siinuslaine loomisel hajub esiplaan näiteks sisse ja välja.

Liikuv üleminek (Wipe)



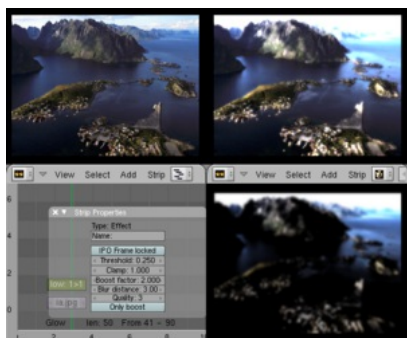
Videomonteerija sisseehitatud liikuv üleminek

Üle pildi liikuv üleminek ühest ribast teise. Sellel väga paindlikul efektil on neli ülemineku tüüpi:

- Kell (*Clock*): üleminek toimub ringjalt nagu analoogkella seierid - kas päripäeva või (kui "Wipe In" on sisse lülitatud) vastupäeva alates kellajast 9:00. Liikudes muutub järgmine riba nähtavaks.
- Iiris (*Iris*): nagu kaamera või silma iiris, muudab see järgmise riba suureneva (või väheneva) ringi abil nähtavaks. Võid selle ülemineku muuta ähmaseks, nii et tulemus näeb välja nagu läbi paberi imbuv tint.
- Topeltpühkimine (*Double Wipe*): alustab joonena keskelt ja liigub väljapoole, muutes järgmise riba nähtavaks. Selle abil saab teha ka vastupidist liikumist (lülitades sisse valiku "Wipe In") - st efekt algab äärtest ning liigub keskkoha suunas. Saad muuta ka liikumissuuna nurka ning seda ähmastada.
- Pühkimine (*Single Wipe*): muudab järgmise riba nähtavaks, liikudes ühest pildi äärest vastasääreni. Seadetega saab muuta liikumise nurka (saad alustada mõnest nurgast või küljest) ja ülemineku ähmastamist.

Märkus: mõned vanemad pluginad pakuvad samaseid võimalusi.

Kuma (Glow)

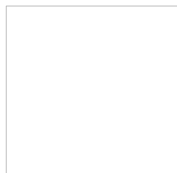


Pildile lisatud kuma efekti (Glow) näidis.

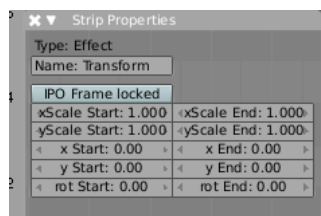
Ülal vasakul: aluspilt (Teravmäed, Norra – allikas: wikipedia.fr);
 Ülal paremal: efekti tulemus;
 All vasakul: efekti sätted;
 All paremal: tulemus, kui sisse on lülitatud nupp Only boost (ainult suurenda).

See efekt paneb pildi heleduskanalit muutes pildi osasid heledamalt kumama. Kuma (Glow) liidab kokku kaks pilti - aluspildi ja selle koopiat, mille heledaid piirkondi (neid, mis on heledamad kui läve (Threshold) väärtus) on udustatud. Kuma riba sätetega saad määrata selle läve (Threshold:), maksimaalse lisatava heleduse (Clamp), helenduse suurendamise kordaja (Boost factor), udustamise suuruse (Blur distance) ja udustamise kvaliteedi (Quality). Nupuga Only boost (ainult võimendus) saad näidata/kasutada ainult 'muudetud' pildiversiooni ilma aluspildita. Kuma efekti "animeerimiseks" sega see aluspildiga gammapõhise üleminekuefekti (Gamma Cross) abil ja loo üleminek aluspildilt helendavale pildile.

Teisendamine (Transform)



See efekt Blenderi versioonis 2.49 ei tööta! Teisendamine on pildi manipuleerimise puhul nagu Šveitsi taskunuga. Selle abil saad muuta ribas olevate piltide mõõtkavasid, neid liigutada ja pöörata. Paremal olevalt näidiselt on näha, mida saab teha üksiku pildiga. Sujuva ülemineku saavutamiseks lülita sisse kaadri lukustamise nupp (Frame locked) ning määra IPO aknas kõver (monteerimisrežiimis).



Kui oled valinud teisendamise riba (Transform), saad kasutada selle efekti juhtimiseks seadete paneeli:

(x,y)Scale (Start,End) (X/Y skaala algus/lõpp)

Määrab mõõtkava (suuruse). xScale Start (X skaala algus) määrab pildi laiuse riba alguses, xScale End (X skaala lõpp) laiuse riba lõpus, yScale Start ja yScale End (Y skaala algus ja lõpp) vastavalt pildi kõrgused riba alguses ja lõpus. Numbrist **1.0** suuremad väärtused suurendavad pilti ning numbrist **1.0** väiksemad vähendavad seda.

(x,y) (Start,End) (X/Y algus/lõpp)

Määrab asukoha (nihutab pilti). x Start (X algus) määrab pildi horisontaalse asukoha riba alguses, x End (X lõpp) riba lõpus; positiivsed väärtused liigutavad pilti paremale, negatiivsed vasakule. y Start (Y algus) määrab pildi vertikaalse asukoha riba alguses, y End (Y lõpp) riba lõpus; positiivsed väärtused liigutavad pilti üles, negatiivsed alla.

rot (Start,End) (pööre algus/lõpp)

Pööre on määratud kraadides (**360** teeb täisringi) ning toimub vastupäeva. Pilti päripäeva pööramiseks, määra pöörde lõpu väärtus väiksemaks kui alguse oma (näiteks alusta väärtusest 360 ja liigu sellest allapoole).

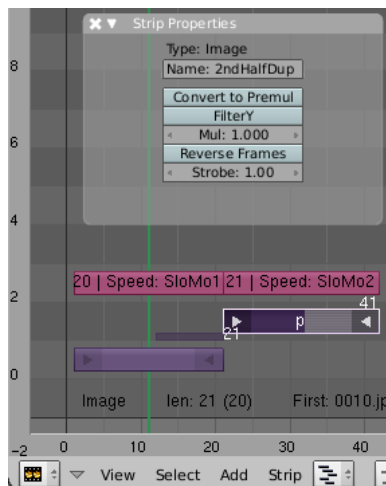
Värv (Color)

See efekt toimib iseseisvalt ning loob värvi riba. Vaikimisi on see loomise hetkel 50 kaadrit pikk, kuid sa saad pikkust otsade lohistamise abil muuta. Vajuta stseeni kaardil (F10) montaažinuppude (Sequencer) all asuvas efektide paneelis värivalijal, et muuta riba värvi (vaikimisi on see hall). Kasuta seda riba koos üleminekuga, et oma peamist filmi sisse või välja hajutada.

Kiiruse määraja (Speed Control)

Kiiruse määraja muudab riba kiirust, pannes selle normaalsest kiiremini või aeglasemalt mängima. Globaalse kiiruse väärtus, mis on madalam kui 1.0, paneb riba aeglasemalt mängima; suurem kui 1.0 paneb riba kiiremini mängima. Kiiremini mängimine tähendab, et mõned kaadrid jäetakse vahele ning ribal saavad kaadrid enne otsa, kui see oma viimase kaadri jõuab. Kui ribal saavad näidatavad kaadrid otsa, jääb see kordama viimast kaadrit - näib, nagu oleks tegevus seiskunud. Selle vältimiseks aseta järgmine riba originaali alla sellesse kohta, kust soovid animatsiooni jätkata.

Aegluubi (Slow Motion) efekti saavutamine



50% aegluubi loomine kiiruse määrajaga

Oletame, et soovid oma riba aaeeglaaseemaaks muuuuutaa. Pead selleks muutma videoklipi kiirust, kuid jätma üldise kaadrisageduse samaks. Vali klipp ning lisa seejärel menüüst Add->Effect->Speed Control (lisa->efekt->kiiruse määraja) kiiruse määraja efekti riba. Klõpsa selle paigaldamiseks hiirenuppu ning ava seaded kiirklahviga N. Määra globaalse kiiruse (Global Speed) väärtusega kordaja, kui palju soovid kiirust muuta. Et vähenda näivat kiirust 50%-ni algsest, sisesta 0.50. Nüüd mängitakse 30-kaadrist klippi poole väiksema kiirusega ning seetõttu näidatakse ainult 15 esimest kaadrit.

Kui soovid, et aegluubis näidatakse ka ülejäänud kaadreid, lõika kiirklahvi K abil klipp kaheks, nihuta teist poolt (sest aegluubis mängimine võtab rohkem aega kui klipi algne pikkus) ning lisa teine kiiruse määraja, nagu on näidatud paremal oleval pildil. Kui kanalil 1 olev riba, mida muudab kanalil 3 olev efekti riba, lõpetab, alustab mängimist kanalil 2 olev riba, mida omakorda muudab kanalil 3 olev kiiruse määraja. Teise riba loomiseks vajalik trikk on järgnev:

- Vali originaalriba ning duplitseeri seda kiirklahviga \diamond ShiftD. Antud juhul on tegemist 20-kaadri pildireaga.
- Lohista duplikaat poolenisti originaalst paremale ja lase siis lahti.
- Vali vasak käepide (algus), haara sellest kinni ning liiguta poole võrra edasi. See muudab alguskaadri nihet - riba alustab mängimist kaadrist 11 (oma sisemise aja järgi);
- Vali parem käepide (lõpp), haara sellest kinni ning liiguta seda samuti poole võrra edasi (antud juhul 10 kaadrit). See määrab, et selle riba pikkuseks sinu videos on 20 kaadrit.
- Lisa esimesele ribale kiiruse määraja väärtusega 50% ja teisele kiiruse määraja väärtusega 100%. Teise riba puhul sa juba ütled Blenderile, et ta mängiks maha 10 kaadrit 20 kaadri aja jooksul, mis tähendabki poolt kiirust - seega ei ole sul vaja kiirusemäärajat, mis seda veelgi aeglustaks, vaid pead lihtsalt muutma kaadrite arvu klipi pikkusele vastavaks.

Ja ongi kõik! Pane oma renderdaja kõigi 40 kaadri (antud näite puhul) animatsiooni renderdama.

Võid küsida, miks me lihtsalt originaalklipi ei pikendanud? Põhjus on lihtne - kiiruse määraja töö põhineb tema poolt näidatavate kaadrite arvul, mis on jagatud kaadrite arvuga, mille jooksul ta seda tegema peab. Kui näiteks ribal on 20 kaadrit ning sa määrad kiirusfaktoriks 50%, siis kiiruse määraja teab, et peab näitama kümnet kaadrit. Kui riba on aga näiteks venitatud 30 videokaadri pikkuseks, siis näitab ta igatüht kümnest kaadrist 3 kaadri jooksul ($3 \times 10 = 30$). Selle matemaatika mõistmine muudab su tegevuse efektiivsemaks ning aitab sul saavutada soovitud.

Väga lihtsatel juhtudel piisab ka originaalklipi pikendamisest. See on väga kasulik tööriist, mis on mõeldud süžeevahvlite tegemiseks ja tühjade kohtade täitmiseks. Lisa kiiruse määraja, ära määra kõveraid, ignoreeri muid parameetreid ja lihtsalt lohistage originaalklipi paremat käepidet edasi. Pea meeles, et see on võimalik ainult siis, kui allolev klipp lõppeb samas kohas, kus asub parem käepide. (Kui see ei ole võimalik, sest sa löid klipi lõikaja abil, siis pead lihtsalt kõigepealt lisama originaalklipi metaklipi).

Kaadrite sobitamine

Oma klipi ajastuse veelgi täpsemaks juhtimiseks saad kasutada kõveraid! Seda saab teha mitmel viisil.

- Vaikimisi seadete puhul laotavad F-kõverad lihtsalt sisendi ümber väljundkaadritele, mille arv on skaleeritud vahemikku [0-100] (vastab F-kõvera koordinaatidele [0-1]). Lihtsate võbelevate efektide loomiseks sellest täiesti piisab.

Kuid kiiruse efekt on suuteline veelgi enamaks. Saad kasutada reaalselt kaadrite sobitamist! See tähendab: sul on olemas kindel asukoht, kus mingit kindlat kaadrit tuleks näidata (näiteks proovid sa sobitada oma videoklipi heliga) ja Blender muudab sujuvalt klipi kiirust, nii et teda täpselt sellel hetkel näidatakse.

- **Kaadrite lukustamise (frame locking)** abil saad keelata mõõtkava muutmise X-suunal. See tähendab: sisendkaadrite numbrid vastavad F-kõvera X-koordinaatidele.
- Y-suunal saad keelata mõõtkava muutmise, lülitades välja valiku "Scaling [0-1]" (skaleerimine [0-1]). See tähendab: väljundkaadrite numbrid on identsed F-kõvera Y-koordinaatidega.

Pärast seda kasuta graafiku aknas olevat N-võtmete (N-keys) akent, et asetada kontrollpunktid täpselt õigetesse kaadritesse.

Video kaadrisageduse muutmine

Kiiruse määrajaga saad muuta video sekundis näidatavate kaadrite arvu (fps) ehk kaadrisagedust. Kui renderdad videot pildijadana, saad suurendada või vähendada loodavate üksikpiltide arvu, kasutades globaalset kiiruse väärtust, mis on vastavalt kas suurem või väiksem kui üks. Kui sa näiteks lindistais vieminutilise video kaadrisagedusega 30 fps ning soovid seda kanda filmilindile (mis jookseb kaadrisagedusega 24 fps), pead sa sisestama globaalse kiiruse väärtuseks 30/24 ehk 1.25 (ning lülitama sisse kaadrite sulandamise (Frame Blending), et saavutada filmile omast udusust). Selle asemel, et tekitada $5 \times 60 \times 30 = 9000$ kaadrit, tekitab Blender $9000 / 1.25 = 7200 = 5 \times 60 \times 24$ kaadrit. Antud juhul määraksid sa alguse (Sta) väärtuseks 1 ja lõpu (End) väärtuseks 7200, väljundi formaadiks Jpeg ja kiiruseks 30 fps. Renderdamise tulemusena luuakse pildifailid 0001.jpg kuni 7200.jpg, kuid need 'katavad ära' kõik 9000 kaadrit. Pildifail 7200.jpg oleks sama, mis kaader 9000. Kui sa loed need pildid uuesti sisse oma 24 fps

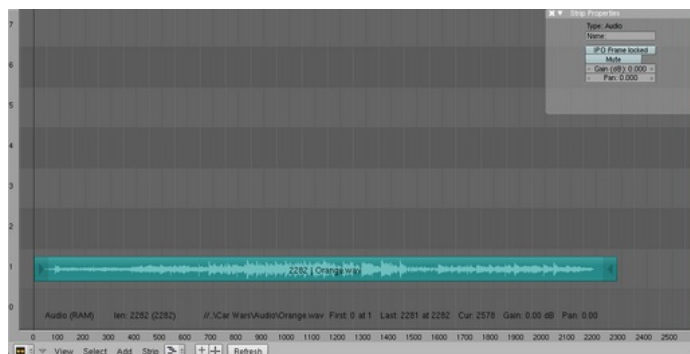
kaadrisagedusega .blend-faili, kestab see riba täpselt 5 minutit.

Mitmikkaamera valija (*Multicam Selector*)

Kohanduskiht (*Adjustment Layer*)

Helimontaaž

Blenderis on olemas mitmerajaline heli monteerimise tööriistakast. Sa saad lisada WAV ja Mp3 formaadis helifaile kas kõvakettalt või videofaili kodeerituna ja segada neid, kasutades F-kõveraid helitugevuse määramiseks.



Heliriba montaažihaldajas.

Seaded

Audio-RAM laeb failid mällu ja mängib neid sealt maha. Mällu saad laadida ainult eraldiseisvaid WAV faile. Audio-HD mängib heli maha otse kõvakettalt ning seetõttu ei kasuta failid arvuti mälu. Sellega saad laadida nii eraldiseisvate WAV failide kui ka videofailide heli.

Mõlema puhul luuakse roheline heliriba. Audio-RAM'i puhul luuakse lainevorm, mis näitab roheline riba sees helilaine kuju ja mille mõõtkava vastab riba kõrgusele. Kuna Audio-RAM'i failid loetakse mällu, ei mõjuta audiodfaili muutmine selle mahamängimist ning selleks, et Blender saaks faili uuesti sisse lugeda, pead sa selle uuesti avama.

Kahin, pragin ja prõksumine

Mõned audiofailidest kasutajad on väitnud, et Audio-RAM'i kasutamisel tekib mõnikord kahin. Järelikult peab toimuma mingit sorti dekodeerimine või sümplimine, mis mängimisel müra tekitab ning mida Audio-HD puhul ei kasutata. Kui kuuled prõksumist ja praginat, tähendab see enamasti, et su riistvara ei suuda reaajas mahamängimisega hakkama saada. Need artefaktid ei jõua lõplikult renderdatud animatsiooni (kuid võivad olla kuulda mängurežiimis). Staatiline kahin näib tekkivat alati siis, kui kaks või enam audioriba ajateljel kattuvad...

Helimontaaž videomontaažihaldajas

Võid kasutada nii palju heliribasid kui soovid ning tulemuseks on kõigi nende kokkusegamine. Igale ribale saab kiirklahviga N avanevast menüüst anda nime ning määrata tema võimendatuse (Gain) detsibellides (dB). Sealtkaudu saad ka riba heli välja lülitada või stereokanalite tasakaalu muuta (Pan): -1 on täielikult vasakul, +1 täielikult paremal, vahepealsed väärtused vastavalt suhtele.

Kattuvad ribad segatakse automaatselt animatsiooni töötlemise ajal kokku. Näiteks võib sul teadustaja hääl olla kanalis 5, taustamuusika kanalis 6 ja taustahelid kanalis 7.

Audioradadega töötamine

Audiorada (riba) on nagu iga teine riba videomonteerijas. Sa saad seda valida ja liigutada, muuta selle alguse nihet, vajutades RMB noolega käepidemete peal, ja lõigata seda osadeks kiirklahviga K. Kasulik näide on kõnest "eeee"-hääliitsuste ja pauside välja lõikamine.

Audioraja omaduste animeerimine

Sa tahad määrata mõne väärtuse 0.0 ja 1.0 vahele ning seada helitugevuse sellele vastavalt (näiteks 0.6 on 60%). Heli saab võimendatust (*Gain*) lisada riba omadustes (N). Võid luua mitmest punktist koosneva kõvera, mis muudab helitugevust klipi mängimise ajal. Kõvera muutmiseks vajuta ⇐ Tab nagu iga teise bezier' F-kõvera puhul.

Y-telg tähistab helitugevust - 1.0 on täistugevus ja 0.0 täielik vaikus. Praegusel hetkel suudab ainult FFMPEG-väljundüsteem heli ja video ühte väljundvoogu kokku segada. Kontrollpunktide lisamiseks vajuta Ctrl LMB ning kõvera muutmiseks ⇐ Tab.

Audioriba animeerimine mõjutab selle helitugevust lõplikus komposiidis. Kasuta audioriba animeerimist, et taustamuusikat sujuvalt sisse/välja tuua või helitugevuse tasemeid sättida. Kihiti asetsevad audioribad liidetakse omavahel kokku; allpool asuv kanal ei lülita kõrgemal olevaid kanaleid välja. See teeb Blenderist audiomikseri. Lisades audioradasid ning kasutades iga raja helitugevuse määramiseks kõveraid, on sinu kasutuses dünaamiline mitmerajaline helimiksija!

Väljund

Väljundiks on seega videofail, kui kasutada nuppu ANIMATION (animatsioon) stseenikonteksti (*Scene*) renderdamise alamkontekstis (*Render*) renderduspaneelil (Render). Audiodfaili saab luua nupu MIXDOWN (miksimine) abil stseenikonteksti (*Scene*) heli alamkontekstis (*Sound*) montaažipaneelil (Sequencer). Nii loodud WAV fail sisaldab lõplikku monteeritud heli ja see luuakse videofailiga samasse kataloogi ja sama nimega kui videofail, kuid laiendiga .WAV. Videot ja audiot saad hiljem mõne teise programmiga kokku miksida või näiteks pildijada ribale lisada, nagu seda kirjeldati eespool.

Blenderi montaažihaldaja eeliseks on lihtsamini saavutatav sünkroonusus tänu kaadrite ja heli töötlemisele samas programmis.

Audio sünkroniseerimise sisselülitamiseks pärast audioraja importimist kasuta nuppude aknas nuppu Scene (stseen, F10) ja vali sealt helibloki nupp (Sound Block, väike sinine siinuslaine). Siit leiad sünkroniseerimise (Sync) ja kerimise (Scrub) tööriistad.

- Sünkroniseerimine (Sync) lubab Blenderil jätta pildikaadreid 3D-aknas animatsiooni mängides vahele, et püsida sünkroonis reaajas oleva heliga. See annab sulle üldise ettekujutuse oma animatsiooni ajastusest.

- Kerimisega (Scrub) saad igas aknas kaadri tähist lohistada või kaadreid vahetada ning audioklippe mängitakse tolele ajahetkele vastavalt.

Lohistades tegevuseredaktoris kaadri tähist mööda kaadreid, kuuled enam-vähem, millistes kohtades mingid helid kostuvad, et saaksid poose ja kujusid vastavas kaadris muuta.

Blender 2.5 Python 3.2 kasutusjuhend

Sissejuhatus

Tere tulemast Blender 2.5/6 Pythoni kasutusjuhendigi tutvuma!

Python www.Python.org on interpreteeritav, interaktiivne, objekt-orienteeritud programmeerimiskeel. See võimaldab kasutada mooduleid, erandolukordi, dünaamilisi tüüpe, kõrgtaseme andmetüüpe ja klasse. Python ühendab märkimisväärse võimsuse väga selge süntaksiga.

Pythoni skriptid on väga võimas ja paindlik viis Blenderi funktsionaalsust laiendada. Enamikke Blenderi tegevusi saab skriptidega juhtida, sealhulgas animatsioone, renderdamist, importi ja eksporti, objektide loomist ja korduvate tööprotseduuride automatiseerimist.

Blenderiga suhtlemiseks saavad skriptid kasutada tihedalt integreeritud programmeerimisliidest (API - Application Programming Interface). See on veel arendusjärgus, kuid juba piisavalt stabiilne, et seda kasutama hakata. Stabiilses Blender 2.5 versioonis on oodata veel ainult väikesi muudatusi.

Hoiatus



Millalgi versioonide 2.56 ja 2.57 vahel tehti väike, aga oluline muutus Pythoni lisade API-sse: kirjeldav struktuur, mille nimi tavatses olla `bl_addon_info`, on nüüd `bl_info`. **Skriptid, mis ikka veel kasutavad struktuuri `bl_addon_info`, ei ole Blenderi kasutajaeelistuste aknas *User Preferences* lisade lehel *Addons* nähtavad!**

Üldine informatsioon

Skriptide kirjutamisel kasulikud lingid (kõik ingliskeelsed).

- [Blenderi Python API](#)
 - Ametlik programmeerimisliidese dokumentatsioon. Kasuta seda skriptide kirjutamisel teatmikuna.
- [API sissejuhatus](#)
 - Lühike sissejuhatus, mis aitab programmeerimisliidese alustada. Sisaldab näiteid.
- [Kokaraamat](#)
 - Kasulike koodijuppide osa (tuleb veel kirjutada)
- [KKK](#)
 - Korduma kippuvad küsimused ja nende vastused

Lingid, mis käsitlevad oma skriptide avaldamist (kõik ingliskeelsed).

- [Skriptide jagamine](#)
 - Informatsioon selle kohta, kuidas oma skripti jagada ja neid ametlikku Blenderi paketti lisada.
- [Lisade \(*Add-Ons*\) loomine](#)
 - Alates Blenderi versioonist 2.5 Alpha1 levitatakse Blenderi skripti sellel meetodil.
- [Laienduste \(*Extensions*\) projekt](#)
 - Projekt, mis haldab kesket Blenderi laienduste kogu.

Alustamine - Wiki õppetükid

Järgnevad leheküljed asuvad selles Wikis ning õpetavad sind päris algusest kuni keerukate Blenderi Pythoni skriptide mõistmiseni.

- [Tervitus \(*Hello World*\)](#)
- [Konsool](#)
- [Tekstiredaktor](#)
- [Geomeetria](#)
- [Omadused \(*Properties*\) ja ID-omadused \(*ID-Properties*\) ning nende erinevused](#)

Alustamine - välised lingid

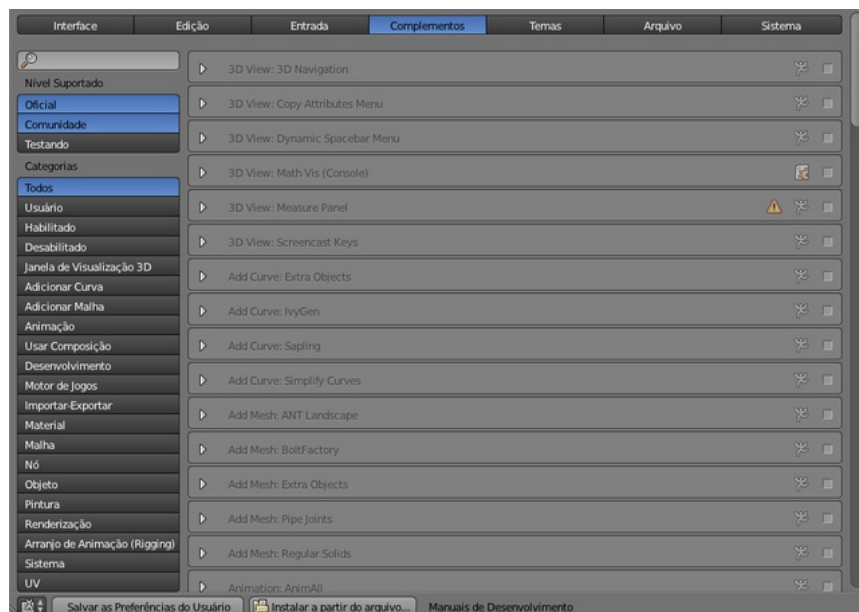
Järgnevad leheküljed ei kuulu sellesse Wikisse, kuid sisaldavad kasulikku õpetusi Blenderi skriptide kirjutamise alustamiseks.

- [Satish Goda sissejuhatav õppetükk](#)
 - Alustab päris algusest ja näitab, kuidas teha põhilisi API operatsioone.
- [Ira Krakowi videoõppetükid](#)
 - Videoõppetükkide seeria esimene video.
- [Kiirjuhend](#)
 - "Kiirjuhend neile, kes Pythonit ja Blenderit juba mingil määral tunnevad.
- [Näidiste jutulõim](#)
 - Foorumi jutulõim, kust leiab palju lühikesi töötavaid skriptinäidiseid.

Erinevalt eelmistest Blenderi versioonidest ei nõua Blender 2.5 täielikku Pythoni installatsiooni. Selle asemel on kõik vajalikud Pythoni moodulid lisatud Blenderi installatsioonikataloogi. \\Blender_2.5\\blender\\python\\lib

Lisad (Add-Ons)

Lisad (Add-On) on üldine termin, tähistamaks valikulisi skripte, mis täiendavad Blenderi võimalusi. Need asuvad kasutaja eelistuste akna User Preferences lisade sakis Add-Ons. Selle saki kaudu saad lisasid installida ning neid sisse ja välja lülitada. Blenderiga tulevad automaatselt kaasa mõned kasulikud lisad, kuid saad lisada ka enda pool tehtuid või huvitavaid internetist leitud lisasid. [Skriptide kataloogis](#) on ära toodud nimekiri Blenderiga kaasas olevatest lisadest ning samuti mitmed välised lisad.



Lisade installeerimine

Et skript lisade sakki ilmuks, tuleb see esmalt installeerida. Selleks saad kasutada lisade akna päises olevat installeerimise nuppu Install Add-On. Vajuta lihtsalt nupule ja otsi üles skript, mida soovid installeerida. Pärast installeerimist ilmub skript koheselt paneeli. Lisasid saad installeerida ka käsitsi. Lisa peetakse installeerituks, kui see asub kataloogis `../scripts/addons` (kus `..` tähistab sinu Blenderi konfiguratsiooni kausta). Lisa töstmisest sellesse kausta (fail, mille laiendiks on `.py`) piisab.

Failide asukohad Blenderi versioonis 2.5

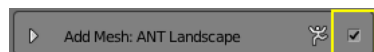
- Windows 7 - `C:\Users\%username%\AppData\Roaming\Blender Foundation\Blender\2.5x\scripts\addons`
- Windows XP - `C:\Documents and Settings\%username%\Application Data\Blender Foundation\Blender\2.5x\scripts\addons`
- Linux - `/home/$user/.blender/$version/scripts/addons`

Pane tähele, et Windows 7-e kataloog `AppData` ja Linuxi kataloog `.blender` on peidetud. Asukoht võib samuti olla erinev, sõltuvalt sinu valikutest nii operatsioonisüsteemi kui ka Blenderi installeerimisel.

Võid luua ka isikliku kataloogi, kus su lisad asuvad, ning määrata selle oma asukoha kataloogiks kasutaja sätete akna User Preferences faili paneelis File. Et luua isiklik skriptide kataloog:

1. Loo tühi kataloog (nt 'skript_lisa_2-5x')
2. Lisa sellesse kataloogi 'lisatud'.
3. Pane oma uued lisad kataloogi 'lisatud'.
4. Ava kasutaja sätete akna User Preferences faili paneel File.
5. Sisesta skripti lahtrisse Scripts oma skriptide kataloogi asukoht (st 'skript_lisa_2-5x').

Sisse ja välja lülitamine



Lisa sisse lülitamine

Pärast installeerimist tuleb lisa enne kasutamist sisse lülitada. Pane lihtsalt soovitud lisa kastis Enable Add-On linnuke ja oledki valmis. Lisa uued võimalused on nüüd Blenderis olemas ja neid saab kasutada.

Lisa välja lülitamiseks võta linnuke kastist ära. Lisa kohta lisainformatsiooni saamiseks saad vajutada sissekandest vasakule jäävat noolt ning sulle näidatakse kogu saadaval olevat lisainfot. Kui lisa sisse lülitamisel ei aktiveeru, kontrolli [konsooli aknast](#), kas laadimisel on toimunud vigasid.



Lisade eelistuse salvestamine

Kui soovid, et mõni lisa oleks Blender käivitamisel iga kord sisse lülitatud, pead salvestama oma [kasutaja eelistused](#).

Arendusjuhendid

Kui sa oled skriptide arendaja, võivad sind huvitada [lisade arendamise juhendid](#).

Miks veel üks Pythoni õppetükk?

See lehekülg eksisteerib kahel põhjusel

1. Et tutvustada kiirelt ja efektiivselt Pythoni versiooniga 3.1 programmeerimist.
2. Ning veelgi olulisemana anda õpetust Blenderi konsooliaknas, et saaksid õppida õiges kontekstis.
3. Vaata allpool olevaid [väliseid linke](#)

Mis on programmeerimine?

Programmeerimine tähendab lihtsalt öelduna andmete manipuleerimist. Andmetega läbi viidud operatsioonid kas muudavad neid või loovad uusi andmeid.

Kõige lihtsamad andmed on numbrid. Numbritega tehtavad operatsioonid on liitmine, lahutamine, korrutamine jms – see on kõige lihtsam võimalik andmete tüüp.

Kuid reaalse maailma probleemide jaoks vajame liitandmeid (*compound data*), mis koosnevad lihtsamatest andmetest nagu numbrid. Heaks näiteks on vektorandmete tüüp, mis koosneb kolmest numbrist.

Selle õppetüki jooksul vaatleme, milliste andmetüüpide kasutamist Pythoni programmeerimiskeel võimaldab, kuidas sa oma programmideks endale sobivaid andmeid saad luua ning kuidas kirjutada operatsioone, mis nende andmetega tegelevad

Mis on Python?

[Python](#) on interpreteeritud, interaktiivne, objekt-orienteeritud programmeerimiskeel. See sisaldab mooduleid, erandeid, dünaamilist tüüpimist, väga kõrge tasemega dünaamilisi andmetüüpe ja klasse. Python kombineerib suure võimsuse väga lihtsa süntaksiga.

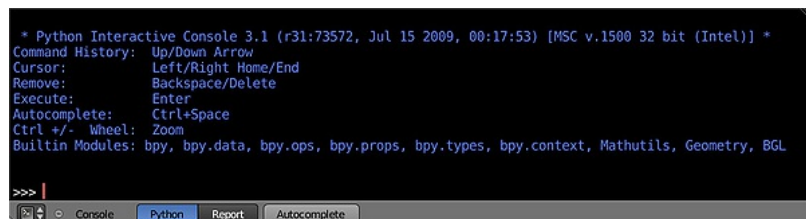
Pythoni õppimine on samas ka väga lihtne, isegi kui sa pole kunagi varem programmeerinud.

Pythoni interpretaator

Kõik selles õpetükis olevad harjutused kasutavad Blenderi versiooni 2.5 sisseehitatud konsooliakna tüüpi, milles sisaldub omakorda Pythoni versiooni 3.1 interpretaator

Järgneb video, mis näitab sulle, kuidas interpretaatorit sisse lülitada

[\[video link\]](#)



Saad Pythoni käske, avaldisi ja lauseid hakata trükima interpretaatori käsureale >>>

Hello World

Alustame klassikalise maailma tervitamise programmiga "Hello World".

Kirjuta interpreteerija käsureale järgnev trükilause ja vajuta klahvi ↵ Enter.



Teeme ülaloleva lause osadeks.

1. "Hello World" on Pythonis stringi [literaali](#).
 1. String on sümbolite jada (numbrid, tähed, erisümbolid)
2. `print()` on Pythonis sisseehitatud funktsioon, mis trükib väljundi.
3. `print("Hello World")` väljastab konsoolile *Hello World*.

Harjutus

trüki järgnevad käsud ja kontrolli väljundit

```
print('Hello World')
print('Hello \n World')
```

Pythonis saab stringiliteraale korrutada *numbriga*. Seda tehes kordame stringi *numbri* poolt nõutud arvu kordi

- number * stringiliteraali
- stringiliteraali * number
- * on Pythonis korrutismärk

```
>>> print("Hello World "*10)
Hello World Hello World Hello World Hello World Hello W
orld Hello World Hello World Hello World Hello World
>>> |
```

Märkus

Vaata [kõiki eelmisi näiteid ühes kohas](#)

Välised lingid

Veebilehed

- <http://www.sthurlow.com/python/>

Videoõppetükid

- Kui sa soovid õppida üldist Pythoniga programmeerimist, vaata seda õppetükki

[\[video link\]](#)

Konsooliredaktori tüüp

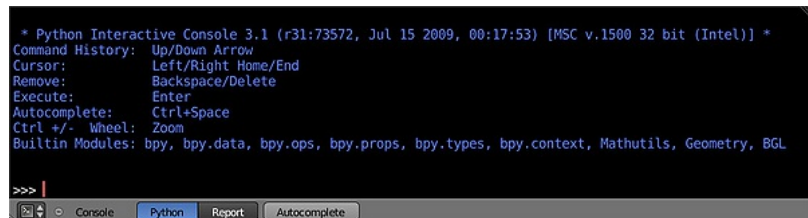
Blenderi versioonis 2.5 täiendati interaktiivset konsooli. Lisati automaatne lõpetamine (*Auto Complete*), Pythoni vigade teavitus (*Python Reporting*) ja palju muid võimalusi. Konsoolis üherealistele käskude testimine on head moodus, kuidas õppida Pythoni programmeerimisliidest (API).

Sisenemine sisseehitatud Pythoni konsooli

Konsooli käivitamine hiirega.

[\[video link\]](#)

Vajutades ükskõik millises Blenderi redaktori tüübis (3D-vaade, ajatelg vms) kiirkombinatsiooni **⇧ ShiftF4**, sisened sa konsooliredaktorisse.

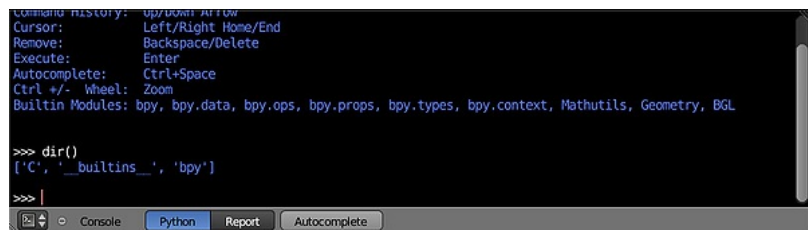


Ülal olevalt ekraanipildilt näed sa, et lisaks tavalistele kiirvalikutele, mida saad navigeerimiseks kasutada, võid sisse lülitada automaatse lõpetamise võimaluse (*Auto Complete*), kasutades kombinatsiooni CtrlSpace.

Kuna Blenderi versioon 2.5 kasutab Pythoni versiooni 3.1, on käskude interpretaator laetud ja valmis käsureal **>>>** sinu käske vastu võtma.

Esmane ülevaade konsooli keskkonnast

Kontrollimaks, mis on interpretaatori keskkonda laetud, kirjuta käsureale dir() ja käivita see klahviga Enter.



Järgnev on väljundi kiire ülevaade

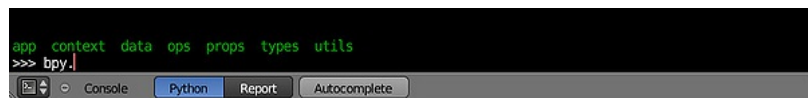
'C'
Blenderi konteksti pide (*Blender's Context handle*)

'__builtins__'
Pythoni versiooni 3.1 sisseehitatud võimalused (klassid, funktsioonid, muutujad)

'bpy'
Kõrgeima taseme Blenderi Pythoni API moodul.

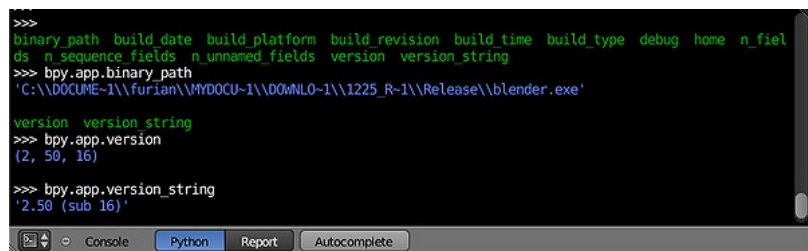
Automaatne lõpetamine töötamas

Nüüd trüki bpy. ning vajuta seejärel CtrlSpace, et näha, kuidas konsooli automaatse lõpetamise võimalus tegutseb.



Näed, kuidas ilmub nimekiri bpy sees olevatest alammodulitest. Need moodulid võtavad kokku kõik, mida me Blenderi Pythoni APIga teha saame, ja need on väga võimsad tööriistad.

Loeme üles bpy.app mooduli sisu.

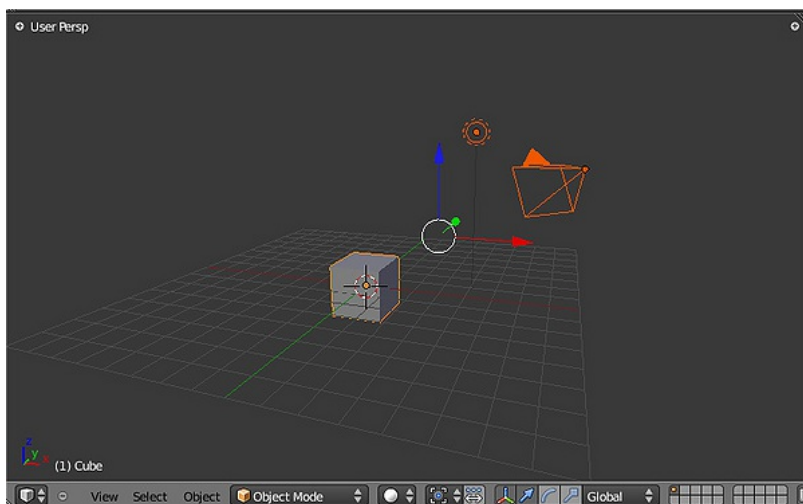


Pane tähele rohelist väljundit selle käsurea kohal, kus sa lülitasid sisse automaatse lõpetamise. See, mida seal näed, ongi automaatse lõpetamise tulemuste nimekiri. Ülal olevas nimekirjas on kõik sissekanded mooduli atribuutide nimed, kuid kui sa näed mõnda nime, mis lõppeb sümboliga '(', siis on tegemist funktsiooniga.

Me kasutame seda võimalust sageli, et oma API õpinguid kiiremaks muuta. Kui sa nüüd seda mõistad, siis liigume edasi ja uurime mõningaid bpy mooduleid.

Enne moodulite kallal nokitsemist...

Kui sa vaatad Blenderi vaikesseeni 3D-vaateaknas, näed sa kolme objekti: kuupi, valgustit ja kaamerat.



- Kõik objektid eksisteerivad mingis kontekstis ning nendega saab tegeleda erinevate režiimide all.
- Igal ajahetkel on ainult üks objekt aktiivne, kuid valitud objekte võib olla rohkem kui üks.
- Kõik objektid on Blenderi failis olevad andmed.
- Neid objekte loovad ja muudavad teatud operaatorid ja funktsioonid.

Kõigi ülaltoodud võimaluste puhul (pea meeles, et me ei toonud ära kõiki võimalusi) on bpy moodul see, mis pakub meile võimalusi andmeid leida ja muuta.

Näited

bpy.context

Märkus

Et allpool toodud käsud näitaksid õiget väljundit, vali kindlasti 3D-vaates objekt(e).

```
>>> active_base active_bone active_object active_pose_bone area driver_add( edit object editable_
bones get( id data is property_hidden( is property_set( items( keyframe insert( keys( main_
manager mode object particle edit_object path_resolve( path_to_id( recast_type( region rna_t
type scene screen sculpt_object selected_bases selected_bones selected_editable_bases selected
editable_bones selected_editable_objects selected_objects selected_pose_bones space_data textu
re_paint_object tool_settings user_preferences values( vertex_paint_object visible_bones visib
le_pose_bones weight_paint_object window
>>> bpy.context.mode
'OBJECT'

>>> bpy.context.object
[BPY_StructRNA "Object" -> "Cube"]

active_base active_bone active_object active_pose_bone
>>> bpy.context.active_object
[BPY_StructRNA "Object" -> "Cube"]

selected_bases selected_bones selected_editable_bases selected_editable_bones selected_editable_
objects selected_objects selected_pose_bones
>>> bpy.context.selected_objects
[[BPY_StructRNA "Object" -> "Cube"], [BPY_StructRNA "Object" -> "Lamp"], [BPY_StructRNA "Object" ->
"Camera"]]
```

Proovi seda!

bpy.context.mode

tükkib ekraanile antud hetke 3D-vaate režiimi (objektirežiim, muutmisrežiim, voolimisrežiim jne)

bpy.context.object või bpy.context.active_object

võimaldab sul kasutada 3D-vaates aktiivset objekti

```
>>> bpy.context.object.location.x = 1
```

Muuda x asukoha väärtuseks 1

```
>>> bpy.context.object.location.x += 0.5
```

Liiguta objekti varasemast x asukohast 0,5 ühiku võrra

```
>>> bpy.context.object.location = [1, 2, 3]
```

Muudab x, y, z asukohta

```
>>> bpy.context.object.location.xyz = [1, 2, 3]
```

Sama, mis eelmine

```
>>> type(bpy.context.object.location)
```

Objekti asukoha andmetüüp (*data type*)


```
>>> dir(bpy.context.object.location)
```

Vaat' siin on alles palju andmeid, mida sa kasutada saad!

bpy.context.selected_objects

annab sulle nimekirja kõigist valitud objektidest (kaasa arvatud aktiivne objekt)

```
>>> bpy.context.selected_objects ja seejärel vajuta {{Shortcut|Ctrl|Space}}
```

```
>>> bpy.context.selected_objects[0]
```

Trükitab nimekirjas esimese objekti nime

```
>>> [object for object in bpy.context.selected_objects if object != bpy.context.object]
```

Keeruleine... Kuid see trükitab nimekirja objektidest, milles ei sisaldu aktiivne objekt

bpy.data

bpy.data sisaldab hunnikut funktsioone ja muutujaid, mille abil saad kasutada kõiki Blenderi failis olevaid andmeid.

Avatud Blenderi failis saad kasutada järgmisi anmeid:

objektid, võred, materjalid, tekstuurid, stseenid, ekraanid (*screens*), helid, skriptid, tekstid, kaamerad, kõverad, valgustid, pintsliid, skeletid, pildid, sõrestikud (*lattices*), teegid (*libraries*), maailmad (*worlds*), grupid, metapallid, osakesed, sõlmegrupid

Neid andmeid on kõvasti.

Proovi seda!

```
BoolProperty( BoolVectorProperty( CollectionProperty( EnumProperty( FloatProperty( FloatVectorP
property( IntProperty( IntVectorProperty( PointerProperty( StringProperty( actions add_image(
armatures bl_rna brushes cameras curves driver add( filename get( gpencil groups id data
images is_property_hidden( is_property_set( items( keyframe insert( keys( lamps lattices lib
raries materials meshes metaballs node_groups objects particles path resolve( path_to_id( r
ecast_type( rna_type scenes screens scripts sounds texts textures values( vfonts window_ma
nagers worlds
>>> bpy.data
[BPY_StructRNA "Main"]
>>> bpy.data.objects
[BPY_PropertyRNA "Main" -> "objects"]
>>>
>>> for object in bpy.data.objects:
...     print(object.name + " is at location " + str(object.location))
...
Camera is at location [7.481132, -6.507640, 5.343665](vector)
Cube is at location [0.000000, 0.000000, 0.000000](vector)
Lamp is at location [4.076245, 1.005454, 5.903862](vector)
>>> |
```

Harjutus

```
>>> for object in bpy.data.scenes['Scene'].objects: print(object.name)
```

vajuta kaks korda ↵ Enter

Trükitab kõigi objektide nimed, mis kuuluvad Blenderi stseeni nimega "Scene"

```
>>> bpy.data.scenes['Scene'].objects.unlink(bpy.context.active_object)
```

Lingib Blenderi stseeni "Scene" aktiivse objekti lahti

```
>>> bpy.data.materials['Material'].shadows
```

```
>>> bpy.data.materials['Material'].shadows = False
```

bpy.ops

Blenderi versiooni 2.5 tööriistades/tegevuste süsteem on ehitatud operaatorite mõistele. Neid operaatoreid saab käivitada otse konsoolist, kuid neid saab käivitada ka nupuvajutusega või pakkida mõna Pythoni skripti. Ned on väga võimsad...

Tekitame 3D-vaateaknasse viis kuupi. Kustuta esmalt olemasolev kuup, vajutades pärast selle valimist klahvi X.

Proovi seda!

Järgnevad käsud määravad, et objektid luuakse kihile 1. Seega, esmalt defineerime hilisemaks kasutamiseks massiivi muutuja:

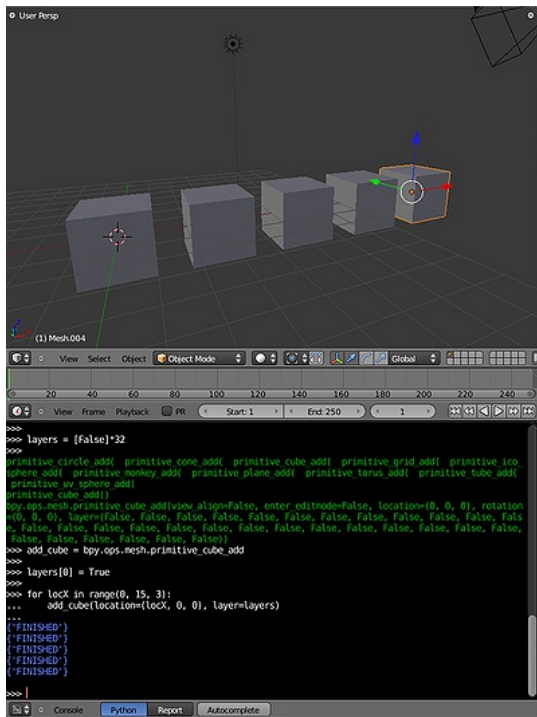
```
>>> mylayers = [False]*20
>>> mylayers[0] = True
```

Looome viite operaatorile, mida kasutatakse kuubi võreprimitiivi loomiseks


```
>>> add_cube = bpy.ops.mesh.primitive_cube_add
```

Kasutades silmust *for loop*, loome järgnevalt viis objekti (all olevas ekraanipildis kasutasin ma üht teist meetodit). Vajuta pärast konsoolireale käsu sisestamist kaks korda Enter-klahvi.

```
>>> for index in range(0, 5):
...     add_cube(location=(index*3, 0, 0), layers=mylayers)
```



Tekstiredaktor

Üks Blenderi aknatüüpidest on tekstiredaktor (Text Editor), mille saab kätte nupuga Text Editor () aknatüübi menüüs või kiirklahviga ⇧ ShiftF11.

Vastavatud tekstiaken on hall ja tühi väga lihtsa tööriistaribaga (*Tekstiakna päis*).



Tekstiakna päis.


Vasakult paremale on siin standardne aknatüübi valiku nupp ja akna menüüd. Seejärel tuleb tekstibloki (*Text ID Block*) valikunupp ja nupp *New*(uus) uute tekstifailide loomiseks. Sellele klõpsates näed, et päis muutub – päris põhjalikult!



Tekstiakna päis avatud failiga

Nüüd on siin väli, millelt saab tekstifaili nime muuta, ja nupp + uute failide tegemiseks. Tekstibloki eemaldamiseks vajuta nupule X.

Järgnevad kolm nuppu lülitavad vastavalt reanumbrite näitamise, ridade poolitamise ja süntaksi esiletõstmise sisse ning välja.

Klaviatuurile kirjutamine loob puhvrise teksti. Nagu tavaliselt, valib LMB -klõps, lohistamine ja vabastamine teksti.

Töötavad järgmised kiirklahvid:

- CtrlC - kopeerib valitud teksti lõikelauale.
- CtrlX - eemaldab valitud teksti ja paigutab ta lõikelauale.
- CtrlV - lisab teksti lõikelaualt tekstiaknasse kursori asukohale.
- ⇧ ShiftCtrlAltS - salvestab muudetud teksti faili avades failisirvija akna (File Browser).
- AltS - salvestab juba avatud faili.
- AltO - laeb teksti failist avades failisirvija akna (File Browser).
- AltP - käivitab teksti Pythoni skriptina.
- CtrlZ - tühistab.
- Ctrl⇧ ShiftZ - tee uuesti.
- AltR - avab (loeb) käsiloleva faili uuesti (kõik salvestamata muudatused lähevad kaduma).
- AltM - konverteerib tekstiakna sisu 3D-tekstiks (maksimaalselt 100 tähemärki).

Tekstipuhvri kustutamiseks vajuta lihtsalt nuppu X tema nime kõrval, nagu sa teed ka materjalidega jms.

Kõige tähelepanuväärsem kiirklahv on AltP, mis edastab puhvri sisu Blenderi sisseehitatud Pythoni interpretaatorile käivitamiseks. Järgmisel leheküljel on Pythonis skriptide kirjutamise näited. Enne jätkamist tasub mainida, et Blenderil on sisseehitatud täisfunktsionaalne Pythoni interpretaator paljude Blenderile spetsiifiliste lisamoodulitega, nagu on lähemalt kirjeldatud [API teatmikus](#).

Tekstiredaktoril endal on nüüd samuti mõned spetsiaalsed Pythoni skriptid, mis lisavad kasulikke kirjutamise abivahendeid, näiteks klasside/funktsioonide/muutujate sirvija, automaatiõpetus jne. Need on kättesaadavad menüüst Text → Text Plugins (tekst -> tekstiplugid).

Tekstiakna teised kasutusvõimalused

Tekstiaken on kasulik ka siis, kui sa tahad oma .blend-faile kogukonna või sõpradega jagada. Selle abil saab kirjutada README teksti, mis annab selgitusi ja juhiseid faili sisu kohta. See on palju mugavam kui selle tegemine eraldi programmiga. Vaata ainult, et see aken oleks nähtav, kui sa faili salvestad! Kui sa jagad faili kogukonnaga ja tahad määrata sellele kasutuslitsentsi, võid sa litsentsi samuti tekstiaknasse kirjutada.

Näide

[\[video link\]](#)

Ülesanne

Kopeeri allpool asuv tekst tekstiredaktorisse.

```
import bpy
from math import radians, cos, sin

# Leia viide kuupvõre lisamise operaatorile

add_cube = bpy.ops.mesh.primitive_cube_add

# Objekt saab asuda 20-l kihil.
# Järgnev koodiosa määrab, millisele kihile sa tahad ta luua.

# Loob nimekirja 20-st elemendist, mille kõigi väärtuseks on "väär"
mylayers = [False]*20

# Määrab esimese elemendi väärtuseks "tõene"
mylayers[0] = True

# Loe kursori asukoht
cursor = bpy.context.scene.cursor_location

# Ringi raadius
radius = 5

# Paiguta kuubid ühtlaselt ringile. Vaikimisi on nad 36-kraadiste nurkade all
# Loo nurkade nimekiri radiaanides
```

```
anglesInRadians = [radians(degree) for degree in range(0, 360, 36)]

# Läbi järjest kõik nurgad, arvuta polaarkoordinaatidest X- ja Y-koordinaadid
# ja loo uus objekt
for theta in anglesInRadians:
    x = cursor.x + radius * cos(theta)
    y = cursor.y + radius * sin(theta)
    z = cursor.z
    add_cube(location=(x, y, z), layers=mylayers)
```

Käivita skript kiirklahviga AltP.

Selles videos on näha, mis peaks juhtuma antud skripti käivitamisel.

[\[video link\]](#)

Sissejuhatus

Kuna me töötame uue ja täiendatud Pythoni programmeerimisliidesega (API), siis palun lisa siia oma küsimused, kui neid tekib. Me uurime arendajatelt vastused välja, kui me neid ise ei tea, ja postitame need siia.

Geomeetria

Kuidas ma saan API abil võreobjekti luua?

Lae endale koodinäide [Script_GeneratePyramidMesh.py](#) ning käivita see tekstiaknast.

Kuidas ma API abil töötleja lisan?

```
bpy.ops.object.convert(target='MESH', keep_original=False)
```

Lisatakse kõik pinus olevad töötlejad.

,Kui sa soovid lisada ainult pinna tükeldamise töötlejat jättes teised rahule, ja luua uut võret (vana võre säilitab kõik oma töötlejad), siis näitab sulle seda järgnev koodijupp.

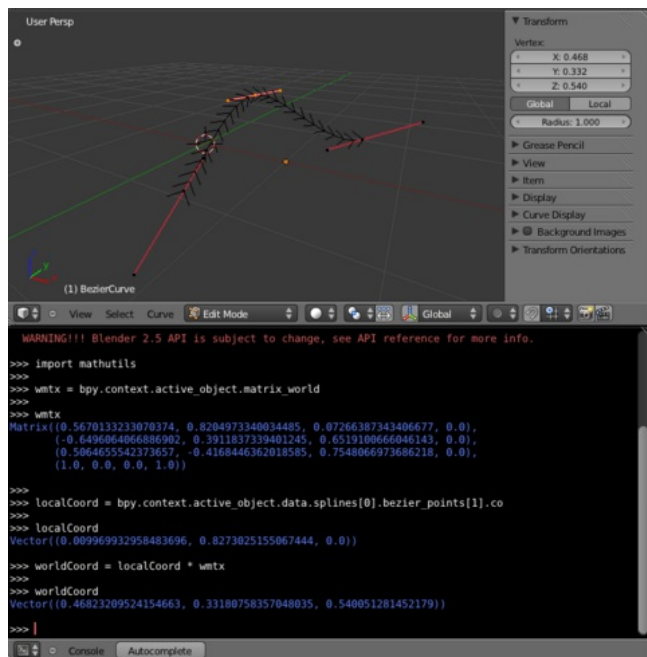
```
for modifier in bpy.context.object.modifiers:
    if modifier.type != 'SUBSURF':
        modifier.show_render=True
bpy.ops.object.convert(target='MESH', '''keep_original=True''')
```

Kuidas ma leian Bezier' kõvera kontrolltipu maailma koordinaadid?

```
wmtx = bpy.context.active_object.matrix_world
```

```
localCoord = bpy.context.active_object.data.splines[0].bezier_points[1].co
```

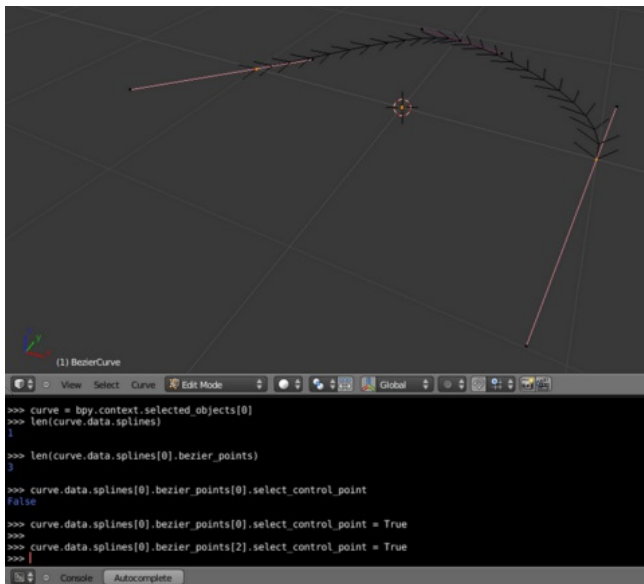
```
worldCoord = localCoord * wmtx
```



[Lisainfo...](#)

Kuidas ma valin kõvera kontrollpunktid või tühistan selle valiku?

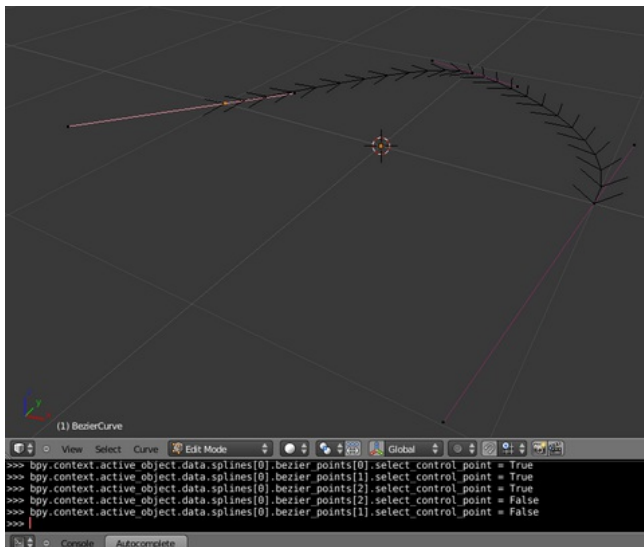
Meetod 1



```
curve = bpy.context.selected_objects[0]
```

```
curve.data.splines[0].bezier_points[0].select_control_point = True
curve.data.splines[0].bezier_points[2].select_control_point = True
```

Meetod 2



```
bpy.context.active_object.data.splines[0].bezier_points[0].select_control_point = True
```

[Lisainfo...](#)

Materjalid

Kuidas linkida võret/objekti materjaliga?

Vaja teha

Ise täiendamine

Kuidas ma endatehtud kiirvalikud automatiseerin?

Blenderi Pythoni API

Blenderi Pythoni API (programmeerimisliides ehk *Application Programmer Interface*) on dokumenteeritud siin:

[2.60a API](#)

[2.59 API](#)

[2.58 API](#)

[2.57 API](#)

[2.56 API](#)

Skriptid

Internetis on Blenderi jaoks saadaval enam kui sada erinevat skripti.

Nagu pluginadki, on skriptid väga dünaamilised, muutes oma kasutajaliidest, võimalusi ja asukohta veebis väga kiirest, seega värske nimekirja ja uusimate linkide jaoks vaata ühte kahest peamisest Blenderi leheküljest:

- www.blender.org
- www.blenderartists.org
- [Selles vikis olevad Pythoni laiendid.](#)

Blenderi mängumootori ülevaade

Blenderisse on sisse ehitatud oma mängumootor, mis võimaldab luua interaktiivseid 3D-rakendusi. Blenderi mängumootor on võimas ja kvaliteetne programmeerimistööriist. Peamiselt on see küll mõeldud mängude loomiseks, kuid mängumootorit võib kasutada ka muu 3D-tarkvara, näiteks interaktiivsete arhitektuuriliste 3D-ekskursioonide või füüsika õppematerjali loomiseks.

Mängumootori kasutamine

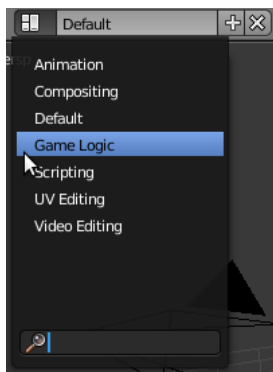
Mängumootori tuumaks on loogikaklotsid ([Logic Bricks](#)). Loogikaklotside ülesandeks on kergesti kasutatava visuaalse liidese kaudu võimaldada interaktiivsete rakenduste kavandamist, omamata sealjuures vähimatki ettekujutust programmeerimisest. Loogikaklotse on kolme sorti: sensorid, kontrollid ja käitid. Täpsemalt on neid kirjeldatud siin:

- [Sensorid \(Sensors\)](#)
- [Kontrollid \(Controllers\)](#)
- [Käitid \(Actuators\)](#)

Kui sa eelistad kirjutada mängu Pythonis, siis mängumootoril on selleks oma eraldiseisev [Pythoni API](#), mida saad kasutada mängu juhtimisskriptide kirjutamisel. Selleks luuakse Pythoni kontroll ([Python Controller](#)) ja seotakse see Pythoni skriptiga.

Mängu loogika

Mängu loogika on kõige mängus toimuva alus. See on võimas töörist loogika loomiseks graafilise kasutajaliidese abil. Klotsid (või "tellised") on eelprogrammitud funktsioonid, mida ühendades ja timmides mängu/rakendusi luuakse. Süsteem on jaotatud kolme ossa: [sensorid](#), [kontrollerid](#) ja [käitured](#). Sensor tajub, kui midagi juhtub. Nagu pörkumine, klahvi vajutus, hiire liikumine. Sensorid on [seotud](#) kontrolleritega, mis võrdlevad neid ja käivitavad käitured.

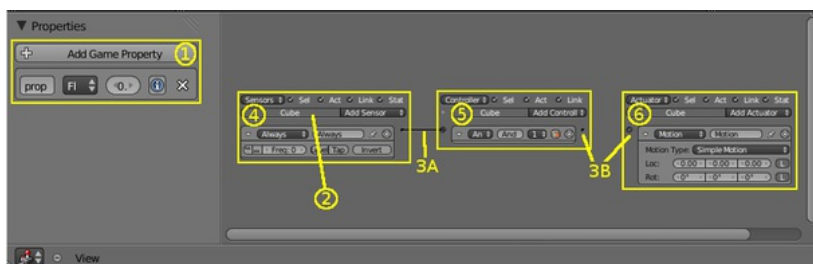


Avades esmakordselt loogikapaneeli (konteksti), vajutades Ctrl→ või valides töökeskkonna ülesehituste menüüst mänguloogika (*Game Logic*).



Loogikapaneel.

Loogikapaneeli paremaks mõistmiseks on osad alloleva pildi menüüdest suurendatud ja nummerdatud. Vaatame igat lõiku eraldi vastavalt numbridusele.



Loogikapaneeli erinevad osad.

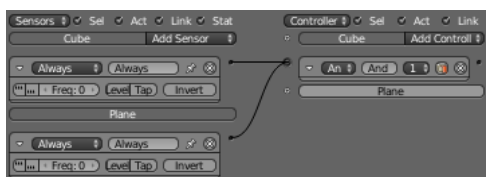
1 Omadused (*Properties*)

Omadused on nagu muutujad teistes programmeerimiskeeltes. Neid kasutatakse objektiga seotud parameetrite lugemiseks ja muutmiseks. Omadusi on järgnevat tüüpi:

- Timer (taimer) - algab määratud numbrist ja kasvab suuremaks.
- String (tekst) - sisaldab teksti.
- Float (ujukomaarv) - hoiab kümnendarve -10000.000 ja 10000.000 vahel.
- Int (täisarv ehk *Integer*) - hoiab täisarve -10000 and 10000 vahel.
- Bool (kahendmuutuja) - on kas tõene (*True*) või väär (*False*).

Põhjalikumaks tutvumiseks vaata [omadusi](#).

2 Seotud objekt(id) (*Associated Objects*)



Mitme objekti loogika.

Mängumootori loogika on seotud objektidega. Objektid järjestuvad nime järgi ja ilmuvad valimisel loogika aknasse. Võid valida üksiku objekti nagu ülaloleval pildil või mitu nagu parempoolsel pildil.

Objekti loogika on nähtav ainult juhul, kui selle loogikaga seotud objekt(id) on valitud.

3 Lingid (*Links*)

Lingid (3A) on objektidevahelise loogikavoo juhtijad. Linkide jooni tõmmatakse hiireklahviga LMB ühest lingisõlmest (3B) teiseni lohistades. Linke saab tõmmata ainult sensorist kontrolleriini või kontrolleriist käituriini. Saatvad sõlmed (mustad ringid sensorite ja kontrolleriite paremal küljel) võivad saata mitmesse vastuvõtvasse sõlme (mis asuvad kontrolleriite ja käituriite vasakul küljel). Vastuvõtvad sõlmed võivad samuti võtta vastu mitmeid linke.

Sensoreid ei saa käituriitega otse ühendada. Käituriid ei saa tagasi ühendada sensoritesse. Kui sa tahad sensorit pärast käituri lõpetamist aktiveerida, siis kasuta käituri sensorit (*actuator sensor*).

4 Sensorid (*Sensors*)

Sensorid alustavad kõiki loogilisi tegevusi. Sensorid "tajuivad" asju. Näiteks lähedal asuvaid objekte, klahvivajutusi, ajastatud sündmusi jne. Kui sensor on käivitunud, saadetakse impulss kõigile ühendatud kontrolleriitele.

Põhjalikumaks tutvumiseks vaata [sensorite osa](#).

5 Kontrollerid (*Controllers*)

Kontrollerid reguleerivad loogikat, hindavad sensorite impulsse ja saadavad impulsse käituriitele edasi. Kontrollerite tüübid on:

- AND (ja) - positiivne impulss saadetakse siis, kui kõigi ühendatud sensorite väljundid on tõesed.
- OR (või) - üks või mitu ühendatud sensorit peavad olema tõesed.
- XOR (välistav või) - üks ja ainult üks ühendatud sensor peab olema tõene.
- NAND (mitte ja) - tagurpidi AND kontroller.
- NOR (mitte või) - tagurpidi OR kontroller.
- XNOR (välistav mitte või) - välistav NOR kontroller.
- Expression (avaldis) - kirjuta oma avaldis.
- Python - juhib sensorit Pythoni skripti või mooduliga.

Põhjalikumaks tutvumiseks vaata [kontrollerite osa](#).

6 Käituriid (*Actuators*)

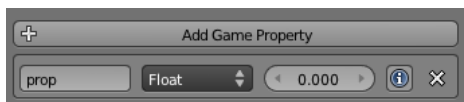
Käituriid mõjutavad objekte mingil viisil. Käituriid muudavad liikumist, heli, omadusi, objekte jne. Need muutused võivad kajastuda teistes objektides, füüsikas, omadustes või käivitada sündmusi teiste loogikaklotside jaoks.

Põhjalikumaks tutvumiseks vaata [käituriite osa](#).

Omadused (*Properties*)

Omadused on mänguloogika vaste muutujatele. Neid talletatakse objektides ja kasutatakse nendega seonduvate asjade, näiteks laskemoona, tervise, nime jne esitamiseks.

Omaduste paneel



Omaduste paneel.

Ava omadused, valides View » Properties (vaade->omadused) või vajutades kiirklahvi N

Omaduste paneel on jaotatud kuueks osaks:

Omaduse lisamise nupp (Add Property)

See nupp lisab nimekirja uue omaduse. Vaikimisi on selleks ujukomaga omadus (Float) nimega "prop", millele järgneb number, kui sellise nimega omadus juba eksisteerib.

Väli Name (nimi)

Siin antakse omadusele nimi, mille kaudu pöördutakse selle poole läbi Pythoni või avaldiste. Pythonis tehakse seda sõnastiku stiilis teatmetabelina (`GameObject["propname"]`). Nimi on tõstutundlik.

Omaduse tüübi menüü (*Property*)

Määrab, mis tüüpi omadusega on tegu ([loe altpoolt](#)).

Väärtuse väli (*Value*)

Määrab omaduse algväärtuse.

Vigade otsimise nupp *Debug*

Lülitab sisse vigade otsimise (*debugging*). Ära unusta panna linnukest ka vigade otsimise omadusi näitavasse lahtrisse (Show Debug Properties) mängumenuis (Game). Kui see on tehtud, näidatakse mängimise ajal kõiki vigasid nende objektinime, omadusenime ja väärtusega. See on vajalik, kui kahtlustad, et mõni su määratud omadustest põhjustab probleeme.

Kustutamisenupp

Iga omaduse kõrval on nupp selle kustutamiseks.

Omaduste tüübid

Omadusi on viit tüüpi:

Timer (Timer)	Alustab omaduse väärtusest ja kasvab suuremaks, kuni objekt eksisteerib. Kasutamiseks näiteks juhul, kui soovid teada, kaua kulub mängijal aega taseme läbimiseks.
Float (ujukomaarv)	Kasutab väärtusena kümnendarvu vahemikus -10000.000 kuni 10000.000. Kasulik täpsete väärtuste hoidmiseks.
Integer (täisarv)	Kasutab väärtusena täisarve (terveid numbreid), -10000 ja 10000 vahel. Kasulik näiteks laskemoona loendamiseks, kus kümnendarvud on üleliigsed.
String (tekst)	Kasutab väärtusena teksti. Mahutab 128 tähemärki.
Boolean (kahendmuutuja)	Kahendmuutuja, tõene (<i>true</i>) või väär (<i>false</i>). Kasulik asjade puhul, millel on kaks olekut, näiteks nagu lülit.

Omaduste kasutamine

Loogikareaktoris on nii omaduse sensor kui ka omaduse käitur.

Omaduse sensor (*Property Sensor*)

Omaduse sensoril (Property) on neli režiimi:

Equal (võrdne)

Saadab positiivse impulsi, kui omaduse väärtus võrdub sensoris määratud väärtusega.

Not Equal (erinev)

Saadab positiivse impulsi, kui omaduse väärtus erineb sensoris olevast väärtusest.

Interval (vahemik)

Saadab positiivse impulsi, kui omaduse väärtus on sensori Min ja Max väärtuste vahemikus.
 "Suurem kui" jaoks sisesta omaduse nimi Max väljale ja käivitamiseks madalaim number Min väljale.
 "Väiksem kui" jaoks sisesta omaduse nimi Min väljale ja suurim väärtus Max väljale.
 Omaduste kõrvutamiseks võib sisestada ka teiste omaduste nimesid.

Changed (muudetud)

Saadab positiivse impulsi, kui omaduse väärtus muutub.

Üldisema info saamiseks sensorite kasutamise kohta vaata [sensorite osa](#) ja detailsema info saamiseks omaduse sensori Property kohta vaata [vastavat osa](#).

Omaduse käituri (*Property Actuator*)

Omaduse käituri (Property) on neli režiimi:

Assign (määra)

Annab välja Value väärtuse välja Prop määratud omadusele. See saab olla ainult sama objekti omadus.

Add (liida)

Suurendab omaduse väärtust. Vähendamiseks sisesta negatiivne väärtus. Kahendväärtuse puhul loetakse nullist erinevat väärtust (ka negatiivset) tõeseks.

Copy (kopeeri)

Kopeerib teise objekti omaduse käituri omaniku omaduseks.

Toggle (vaheta)

Vahetab nulli üheks ning kõik nullist erinevad numbrid nulliks. Kasulik sisse/välja lülitite puhul. Ei tööta tekstistringi (String) tüüpi omaduste puhul.

Üldisema info saamiseks käituri kasutamise kohta vaata [käituri osa](#) ja detailsema info saamiseks omaduse käituri (Property) kohta vaata [vastavat osa](#).

Sensorid (*Sensors*)

Sensorid käivitavad loogikaahela. Nende aktiveerijaks võib olla lähedalasuv objekt, klahvivajutus, ajastatud sündmus vms. Kui sensor on käivitunud, saadetakse positiivne impulss kõigile temaga ühendatud kontrolleritele.

Sensori vaate valikud

Blenderi sensorid leiab Loogika (*Logic*) paneelilt. Sensorite alampaneeli ülemises osas on neli nuppu: valitud (Sel), aktiivsed (Act), lingitud (Link) ja olek (State). Lisaks on seal menüü, mis avaneb, kui vajutada sensorite alampaneeli pealkirja Sensors (sensorid) peale. See on sensorite näitamiseks/peitmiseks ja võimaldab mittevajalikke sensoreid peita, vajalikke näidata ja lihtsustab nende leidmist. Sensorid on koondatud objektide kaupa. Iga valitud objekt esineb loetusel ja iga objekti all on tema sensorid.

Menüü Sensors (sensorid) näeb välja nagu silt, aga on tegelikult menüü. Seal on neil valikut:

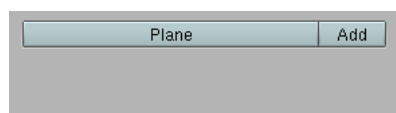
Show Objects (näita objekte)	Näitab kõikide objektide seadeid.
Hide Objects (peida objektid)	Peidab kõikide objektide seaded, jättes igaühest nähtavaks vaid riba tema nimega.
Show Sensors (näita sensoreid)	Näitab kõikide sensorite seadeid.
Hide Sensors (peida sensorid)	Peidab kõigi sensorite seaded, jättes igaühest näha vaid riba tema nimega.

Mõlemaid tegevusi saab teha ka üksikshaaval nii sensorite kui ka objektide puhul.

Võimalik on ka valida, milliseid sensoreid näidatakse:

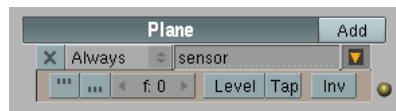
Sel (valitud)	Näitab kõikide valitud objektide sensoreid.
Act (aktiivsed)	Näitab ainult aktiivsete objektide sensoreid.
Link (lingitud)	Näitab sensoreid, mis on mõne kontrolleri lingitud.
State (olek)	Näidatakse ainult hetkeoleku kontrolleri lingitud sensoreid.

Üldised seaded



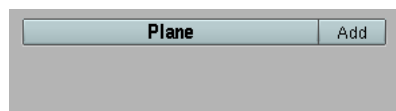
Objekt enne sensori lisamist

Igal objektil on kaks nuppu. Ühel on objekti nimi, teisele on märgitud Add (lisa).



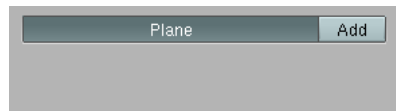
Objekt pärast sensori lisamist

Menüü Add lisab objektile uue sensori. Vaata peatükki [sensorite tüübid](#)



Kokku lükatud sensoritega objekt

Objekti nimega nupp on objekti seadete ja selle sensorite peitmiseks.



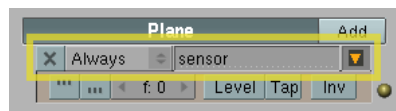
Objekt pärast kõigi sensorite kustutamist

Sensoriteta objekti seaded on automaatselt kokku lükatud. Sensori lisamine näitab automaatselt objekti.

Üldised sensori seaded

Kõigil sensoritel on hulk ühiseid nuppe, välju ja menüüsid. Need on koondatud kahte ritta: **sensori päis** ja **sensori pulsiriba**.

Sensori päis



Sensori päis

Kolmnurkne nupp

Lükkab sensori seaded kokku.

Sensori tüübi menüü

Vaata [sensorite tüüpe](#).

Sensori nimi

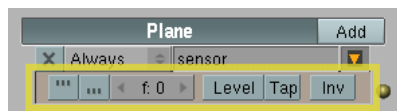
Sensori nimi. Kasutatakse Pythonis sensori poole pöördumiseks; seega peab see valitud objektide hulgas unikaalne olema.

X nupp

Kustutab sensori.

Sensori impulsiriba

Sensorid saadavad impulsse kontrolleritele. Impulsid saavad olla kas TÕESED või VÄÄRAD. Erinevad parameetrid sensori loogikablokis määravad, millal ja millise impulsi sensor saadab. See on äärmiselt oluline, kuna sellel on otsene mõju mänguloogikale ja jõudlusele. Kontrollerid otsustavad, kuidas reageerida TÕENE ja VÄÄR impulssidele.



Sensori impulsiriba

Tõene päästik

Kui see on sisse lülitatud, saavad kontrollerid tõeseid impulsse (TRUE) niikaua, kuni sensori olek on positiivne. Sensor saadab TRUE impulsse vastavalt enda sagedusele.

Väär päästik

Kui see on sisse lülitatud, saavad kontrollerid väärade impulsse (FALSE) niikaua, kuni sensori olek on negatiivne. Sensor saadab FALSE impulsse vastavalt enda sagedusele.

Märkus päästikute kohta

Kui sa päästikuid ei paigalda, ei saada sensorid ühtegi impulssi senikaua, kuni sensori olek ei muutu. Kui sensori olek muutub negatiivsest positiivseks, saadab ta kontrolleritele ühe TRUE impulsi. Kui sensori olek muutub positiivsest negatiivseks, saadab ta kontrolleritele ühe FALSE impulsi.

Vahepeal võivad kontrollerid küll küsida sensori olekut, aga kui kontroller ei saa impulsse (TRUE või FALSE) üheltki teiselt sensorilt, siis ta ise ei aktiveeru.

Frequency (sagedus)

Parameeter "sagedus" ehk **f** (frequency) määrab ajavahemiku korduvate impulsside vahel loogikamootori sammudes.

Vaikeväärtus on 0 ehk viivitamatult.

Loogikamootori sagedus on 60 Hz (60 tiksu sekundis). Näiteks:

- väärtus $f=1$ tähendab, et sensor pulseerib 60 korda sekundis ehk tiksudevaheline suhe on üks ühele. 1 tiks = 1 impulss.
- väärtus $f=30$ tähendab, et sensor pulseerib iga 30 tiksu järel. See tähendab, et impulss saadetakse iga poole sekundi tagant, sest vaikimisi on sekundis 60 tiksu.
- Väärtus $f=60$ tähendab, et sensor pulseerib pärast iga 60. tiksu ehk korra sekundis.

f väärtuse tõstmine on vajalik arvutusvõimsuse säästmiseks, et mootor ei teeks vajalikust rohkem tööd.

Nupp **Level** (tase)

Paneb sensori tööle oleku muutudes. Olekusüsteemi kohta vaata [siit](#).

Nupp **Tap** (koputus)

Saadab positiivse impulsi ainult ühe korra ja seda isegi juhul, kui sensor jääb tõeseks. Aktiveerida saab korraga ainult ühe nuppest **Tap** või **Level**.

Kui **Tap** seade on sisse lülitatud, saadab sensor järgmises kaadris FALSE impulsi, isegi kui sensorit mõjutav sündmus on veel kohal. Kui sensorit mõjutav sündmus kaob, siis impulsse ei saadeta.

Kui *Tõene päästik* on valitud, siis TRUE/FALSE impulsipaar korratakse, kuni sensorit mõjutav sündmus kaob.

Väär päästik seadet ignoreeritakse, kui *Tap* parameeter on valitud.

Impulsse ei pöörata ümber, kui *Inv* parameeter on valitud. Ent TRUE/FALSE impulsipaar saadetakse juhul, kui sensorit mõjutavat sündmust pole kohal.

Nupp **Invert** (ümberpööramine)

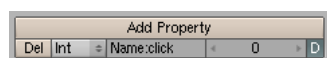
Pöörab sensori väljundi vastupidiseks.

Kui see on valitud, saadab sensor FALSE impulsi, kui ta peaks saatma TRUE impulsi ja vastupidi. Kui *Tap* parameeter on valitud, käitub sensor teisiti (vaata eelmist lõiku).

Märkus *Inv* ja päästikute kohta

Pane tähele, et seade *Inv* pöörab taseme ümber ENNE päästikute käivitamist, mis tähendab, et päästikud reageerivad pööratud väljuvale signaalile.

Testimiseks seadista loogikaklotsid enda valitud objekti ümber alljärgnevalt.



Käivita mäng ja ära vajuta klahvi W: vigadejahtimise info all peaksid sa nägema jooksmas klõpsu (*click*) omadusi, mis muutuvad pärast iga 15. tiksu (veerand sekundit). Seda sellepärast, et sensor on FALSE, aga on pööratud seadega *Inv* TÕESEKS ja seega *Tõene päästik* pulseerib.

[Mänguloogika Pythoni programmiliides.](#)

Sensorite tüübid

Käituri sensor (*Actuator sensor*)



Käituri sensor

Käituri sensor (Actuator) saadab impulsi, kui väljal Actuator (käitur) määratud käitur aktiveerub.

[Mänguloogika Pythoni programmiliides.](#)

Pidev sensor (*Always sensor*)



Pidev sensor

Pidevat sensorit (Always) kasutatakse asjade puhul, mida tuleb teha iga loogikatiksu ajal, iga x-nda loogikatiksu ajal (mitte-null fga) või käivitamisel (Tap abil).

Sellel sensoril pole lisaseadeid peale üldiste.

[Mänguloogika Pythoni programmiliides.](#)

Põrke sensor (*Collision sensor*)



Põrke sensor



Põrke sensor

Põrke sensor (Collision) töötab nagu puutesensor (Touch), aga seda saab ka omaduse kaupa filtreerida. Ainult objektid, millel on vastava nimega omadus, tekitavad põrkumisel positiivse impulsi. Kõikide objektidega põrkumise tuvastamiseks jätta see väli tühjaks.

Nupp **Pulse** (impulss) muudab sensori tundlikuks teiste põrgete suhtes, isegi kui ta on endiselt kokkupuutes selle objektiga, mis päästis valla viimase positiivse impulsi.

Nupp **M/P** vahetab materjali ja omaduste filtreeringut.

Viivituse sensor (*Delay sensor*)



Viivituse sensor

Viivituse sensor (Delay) on mõeldud viivitama reaktsioone määratud arvu loogikatiksude võrra. See on kasulik, kui mõni teine tegevus tuleb teha varem, või sündmuste ajastamiseks.

Viivituse sensoril (Delay) on kolm sätet:

- Delay (viivitus) on loogikatiksude arv, mis sensor enne positiivse impulsi saatmist ootab.
- Duration (kestus) on aeg, mille sensor ootab enne negatiivse impulsi saatmist.
- Repeat (kordus) käivitab sensori uuesti, kui viivituse ja kestuse ajad on läbi.

[Mänguloogika Pythoni programmiliides.](#)

Juhtkangi sensor (*Joystick sensor*)



Juhtkangi sensor



Juhtkangi sensori sündmuste menüü

Juhtkangi sensor (Joystick) reageerib juhtkangi liigutustele.

Indeks (Index) määrab, millist juhtkangi kasutatakse.

Sündmuse tüübi menüü (Event Type) määrab, millist juhtkangi sündmust kasutatakse

Button (nupp)

Määrab, millise numbriga nuppu (button number) kasutatakse.
All Events (kõik sündmused)

Käivitub kõigi selle juhtkangi sündmuste tüüpide peale.

Axis (telg)

Axis Number (telje number) määrab telje
Axis Threshold (telje lävi) määrab täpsusastme
Axis Direction (telje suund) määrab kasutatava suuna: vasak, parem, üles, alla.

Hat (müts)

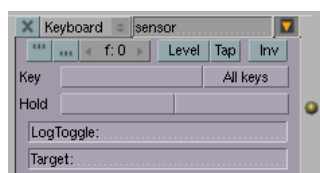
Kasuta määratud mütsi numbrit (Hat number)
Hat Direction (mütsi suund) määrab kasutatava suuna: üles, alla, vasakule, paremale, üles/paremale, üles/vasakule, alla/paremale, alla/vasakule.

Single Axis (üksik telg)

Kasuta ühte telge

[Mänguloogika Pythoni programmiliides.](#)

Klaviatuuri sensor (*Keyboard sensor*)



Klaviatuuri sensor

Klaviatuuri sensor (Keyboard) on klaviatuuri sisendi registreerimiseks ja võimaldab ka klaviatuuri sisendit [teksti tüüpi omadusse](#) salvestada.

Klahvi väli (Key) on üksikute klahvivajutuste jaoks. Vajuta kirjeta nupule ja sellele klahvile, millega tahad sensori ühendada. See on aktiivne klahv, mis vallandab positiivse impulsi. Klahvi määratluse tühistamiseks vajuta nuppu ja seejärel nupust väljapoole.

Kõigi klahvide nupp (All keys) peidab kõik ülejäänud nupud ja saadab positiivse impulsi iga klahvivajutuse peale. See on käepärane [Python kontrolleri](#) abil spetsiaalsete klaviatuurilaotuste loomiseks.

Lisaklahvi nupuga (Modifier) määratud klahvid töötavad samaselt teiste klahvidega, aga saadavad positiivse impulsi ainult siis, kui neid võtmeklahviga samal ajal all hoitakse. Selliseid nuppe saab sul olla kuni kaks. Kui siia on lisaklahv määratud, saadetakse positiivne impulss ainult juhul, kui mõlemad/kõik klahvid on alla vajutatud. See on vajalik juhul, kui soovid näiteks, et CtrlR või ⇧ ShiftAltEsc teevad midagi spetsiifilist.

Logimise väli (LogToggle) määrab kahendväärtusena, kas klahvivajutused salvestatakse tekstimuutujana (String) (tõene) või mitte (väär). See on vajalik, kui soovid logida oma klahvivajutusi.

Sihtmärgi väli (Target) tähistab seda tekstitüüpi omadust, kuhu klahvivajutused salvestatakse. Koos omaduse sensoriga (Property) võib seda kasutada näiteks salasõnade sisestamiseks.

[Mänguloogika Pythoni programmiliides.](#)

Teate sensor (*Message sensor*)

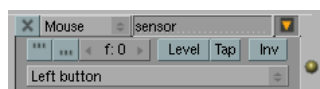


Teate sensor

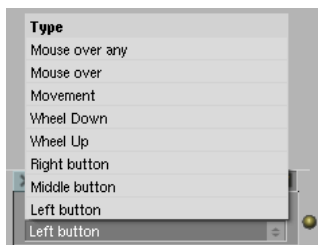
Teateid saab kasutada kas teksti või omaduse väärtuste saatmiseks. Teate sensor (Message) saadab positiivse impulsi siis, kui mängumootoris saadetakse kuhugi sõnum. Seda saab filtreerida saatma impulssi ainult kindlal teemal sõnumi peale.

[Mänguloogika Pythoni programmiliides.](#)

Hiire sensor (*Mouse sensor*)



Hiire sensor



Hiire sensori sisendi tüüp

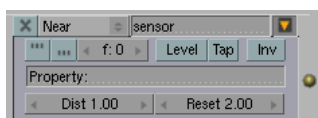
Hiire sensor (Mouse) on hiire sisendi registreerimiseks. Kontrolleri koosneb ainult hiire toimingute loetelust. Need on:

- Left button (vasak klahv).
- Middle button (keskmine klahv).
- Right button (parem klahv).
- Wheel Up (ratas üles), hiire kerimisratas.
- Wheel Down (ratas alla), hiire kerimisratas.
- Movement (liikumine), iga liigutus hiirega.
- Mouse over (hiir objekti kohal), tuvastab, kas hiir on kindla objekti kohal.
- Mouse over any (hiir ükskõik mille kohal), tuvastab, kas hiir on mõne objekti kohal.

Spetsiifiliste hiireliigutuste ja nende mõjude jaoks ei ole eraldi loogikaklotsi (nagu esimese isiku kaamera), vaid need tuleb Pythoniga kodeerida.

[Mänguloogika Pythoni programmiliides.](#)

Läheduse sensor (*Near sensor*)



Läheduse sensor

Läheduse sensor (Near) tuvastab üksteisest määratud kaugusel olevaid objekte. See oskab välja filtreerida kindla omadusega objekte nagu põrkesensor (Collision).

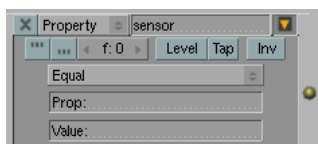
Omaduse väljaga (Property) saab piirata sensorit ainult selle omadusega objekte otsima.

Kaugus (Distance) on vahemaa Blenderi ühikutes, mille piirides objekte tuvastatakse.

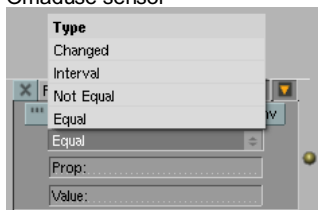
Lähtesta (Reset) on see kaugus, kus objekt peab olema, et sensor lähtestataks (saadaks negatiivse impulsi).

[Mänguloogika Pythoni programmiliides.](#)

Omaduse sensor (*Property sensor*)



Omaduse sensor



Omaduse sensori toimingu tüüp

Omaduse sensor (Property) tuvastab muudatusi objekti omadustes. Sellel sensoril on neli režiimi:

Equal (võrdne) vallandab positiivse impulsi, kui omaduse väärtus võrdub sensoris määratud väärtusega. Omaduse väli (Prop) määrab selle omaduse nime ja väärtuse väli (Value) määrab väärtuse, millega võrdumise puhul saadetakse positiivne impulss.

Not Equal (mitte võrdne) vallandab positiivse impulsi, kui omaduse väärtus erineb sensoris määratud väärtusest. Sellel on samasugused väljad nagu võrdse režiimi (Equal) puhul, kuid väärtus (Value) tähistab väärtust, millest omadus peab erinema, et tekitada impulss.

Interval (vahemik) vallandab positiivse impulsi, kui omaduse väärtus jääb sensori minimaalse (Min) ja maksimaalse (Max) väärtuse vahele. "Enam kui" jaoks sisesta omaduse nimi maksimumi väljale (Max) ja madalaim vallandumiseks vajalik number miinimumi väljale (Min). "Vähem kui" jaoks sisesta omaduse nimi miinimumi väljale (Min) ja kõrgeim väärtus maksimumi väljale (Max). Omaduste võrdlemiseks võib sisestada ka teiste omaduste nimesid.

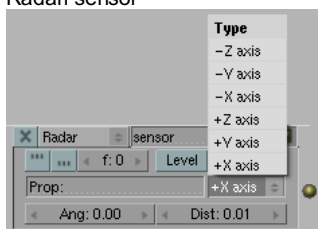
Changed (muudetud) saadab positiivse impulsi kohe, kui omaduse väärtus muutub.

[Mänguloogika Pythoni programmiliides.](#)

Radari sensor (*Radar sensor*)



Radari sensor



Radari sensori telje menüü

Radari sensor (Radar) töötab paljuski nagu läheduse sensor (Near), ainult et nurga all telje suhtes, moodustades nähtamatu koonuse, mille tipp on objekti keskmel ja alus määratud kaugusel piki telge.

Omaduse väljaga (Property) saab piirata sensorit ainult selle omadusega objekte otsima.

Telje menüü (Axis) määrab radari koonuse suuna. \pm märgid otsustavad, kas see on telje suunas (+) või vastupidi (-).

Nurk (Angle) määrab koonuse laiuse.

Kaugus (Distance) määrab koonuse pikkuse.

See sensor on kasulik näiteks selleks, et panna vastased nägema ainult enda ette. Ära unusta, et ta näeb läbi teiste objektide.

[Mänguloogika Pythoni programmiliides.](#)

Juhuslikkuse sensor (*Random sensor*)

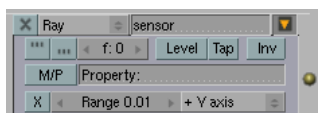


Juhuslikkuse sensor

Juhuslikkuse sensor (Random) genereerib juhuslikke impulsse. Tal on seemne väli (Seed) esmase seemne sisestamiseks. 0 ei ole juhuslik, vaid testimiseks ja vigade otsimiseks.

[Mänguloogika Pythoni programmiliides.](#)

Kiire sensori (*Ray sensor*)



Kiire sensor (lülita pörget omadusega)



Kiire sensor (lülita pörget materjaliga)



Kiire sensori telje menüü

Kiire sensor (Ray) tulistab valitud telje suunas kiire ja saadab positiivse impulsi, kui see midagi tabab. Seda võib filtreerida tuvastama ainult etteantud materjali või omadusega objekte.

Paljud selle nupud ja väljad kattuvad radari sensori (Radar) omadega.

Nupp **X-Ray Mode** (röntgenirežiim) annab läbivalgustava võime, nägemaks läbi objektidest, millel pole filtreerimisväljal määratud omadust või materjali.

[Mänguloogika Pythoni programmiliides.](#)

Puute sensor (*Touch sensor*)



Puute sensor

Puute sensor (Touch) saadab positiivse impulsi, kui objekt puutub kokku teise objektiga. Materjali väli (MA) filtreerib materjale. Positiivse impulsi tekitab ainult kontakt siin väljal oleva materjaliga. Kõikide objektide puutumise tuvastamiseks jäta see tühjaks. Põrkumisel saadetakse positiivne impulss ja kui objektid ei puuduta enam üksteist, saadetakse negatiivne impulss. Kui soovid katkematut impulsijada, kuni objektid on kontaktis, kasuta seadet "True Pulse triggering" (tõene päästik).

[Mänguloogika Pythoni programmiliides.](#)

Kontrollerid (*Controllers*)

Kontrollerid on klotsid, mis koguvad sensorite poolt saadetud andmeid. Sisendi töötlemiseks on kaheksa meetodit:

- [AND \(ja\)](#)
- [OR \(või\)](#)
- [XOR \(välistav või\)](#)
- [NAND \(mitte ja\)](#)
- [NOR \(mitte või\)](#)
- [XNOR \(välistav mitte või\)](#)
- [Avalid \(*Expression*\)](#)
- [Python](#)

Kui sensor aktiveeritakse, saadab see välja positiivse impulsi, ja kui deaktiveeritakse, siis negatiivse. Kontrolleri ülesanne on neid impulsse jälgida ja kombineerida, et vallandada õige reaktsioon.

Kontrollerite võrdlev tabel

Selle tabeli abil saab kiire ülevaate erinevat tüüpi kontrolleritest. Esimene tulp ehk sisend näitab ühendatud sensoritest saabuvate positiivsete impulsside arvu. Järgnevad tulbad näitavad iga kontrolleri vastust neile impulssidele. Vastus "tõene" tähendab, et kontrolleri nõutavad tingimused on täidetud ja sellega ühendatud käitur käivitub. Vastus "väär" tähendab, et kontrolleri nõutavad tingimused pole täidetud ja midagi ei juhtu. Kontrollerite detailsemate kirjelduste saamiseks uuri alljärgnevat lõike.

Märkus

Eeldatakse, et kontrolleriga on ühendatud rohkem kui üks sensor. Ainult ühe sensori puhul vaata rida "Kõik".

Positiivsed sensorid	Kontrollerid					
	AND	OR	XOR	NAND	NOR	XNOR
Puudub	Väär	Väär	Väär	Tõene	Tõene	Tõene
Üks	Väär	Tõene	Tõene	Tõene	Väär	Väär
Mitu, mitte kõik	Väär	Tõene	Väär	Tõene	Väär	Tõene
Kõik	Tõene	Tõene	Väär	Väär	Väär	Tõene

Kontroller AND (ja)

Kui sa lood uue kontrolleri, siis vaikimisi on selle tüübiks AND. Seda sellepärast, et tegemist on kõige levinuma kontrolleriga, mis muu hulgas võib saata sensori signaali lihtsalt otse käituri (mida ei saa ilma kontrollerita teha).

Kontroller AND aktiveerib endaga ühendatud käituri vaid siis, kui kõik ühendatud sensorid on jaatavas seisundis. Kui ühendatud on ainult üks sensor, aktiveeritakse käituriid kohe, kui sensor on aktiveerunud. Kui ühendatud on rohkem sensoreid, peavad need kõik olema üheaegselt käivitunud. Sellest ka kontrolleri nimi. Kui kaks sensorit, "*sensor1*" ja "*sensor2*", on ühendatud kontrolleriga, siis mõlemad *sensor1* ja (*AND*) *sensor2* peavad olema samaaegselt aktiivsed.

Näide

Sa soovid teha menüünuppu. Kui mängija vajutab nuppu, saadetakse ta järgmisse stseeni.

Ühenda menüünupuga kaks [hiiresensorit \(Mouse Sensors\)](#): hiire kohalolu Mouse over ja vasak nupp Left button. Mõlemad ühendatakse sama AND kontrolleriga, mis omakorda ühendatakse stseeni määramise käituri (Set Scene). See tähendab, et mängija peab hoidma hiirt menüünupu kohal ja hiire klahvi vajutama, et siseneda järgmisse stseeni.

Kontroller OR (või)

Kontroller OR käivitab käituriid, kui vähemalt üks selle sensoritest on aktiveerunud. Kui kontrolleriga on ühendatud ainult üks sensor, saadab kontroller positiivse impulsi edasi, kui see sensor käivitub. Kui ühendatud on rohkem sensoreid, piisab neist ühe käivitumisest. OR ei ole välistav, mis tähendab, et see töötab ka nagu kontroller AND: kontroller OR edastab impulsi ka siis, kui kõik sensorid on käivitunud. [XOR kontroller](#) on välistav kontroller.

Näide

Soovid, et mängust saaks väljuda nii klahvi Esc kui ka Q abil. Tekita kaks [klaviatuurisensorit \(Keyboard sensors\)](#): üks klahvi Esc ja teine klahvi Q jaoks. Mõlemad ühendatakse OR kontrolleriga, mis on ühendatud [mängu sulgemise käituri](#)ga. Mäng sulgub, kui üks kahest sensorist aktiveeritakse.

Kontroller XOR (välistav või)

XOR on "välistav või" kontroller. Välistav tähendab, et üks "ja ainult üks *sensoritest peab olema positiivne. Kui ainult üks sensor on kontrolleriga ühendatud, käivitab see käituriid, kui sensor saadab positiivse impulsi. Kui kontrolleriga on ühendatud rohkem sensoreid, aktiveerib see oma käituriid, kui emb-kumb neist on positiivne. Kui mõlemad sensorid on positiivsed, siis kontroller ei käivitu.*

Näide

---TEGEMISEL---

Kontroller NAND (mitte ja)

NAND on lühend terminist "Not And" (mitte ja). Loogikas *not* (mitte) muudab tulemuse vastupidiseks, Not And (mitte ja) on seega AND (ja) kontrolleri vastand. AND kontroller käivitab käitured ainult juhul, kui kõik sensorid on sisse lülitatud, NAND kontroller teeb vastupidi. Alati kui vähemalt üks sensoritest on **mittetõene**, käivitatakse käitured. Kui kontrolleriga on ühendatud ainult üks sensor, aktiveeritakse käitured juhul, kui sensor ei ole sisse lülitatud (kontroller käitub nagu lihtne NOT (ei) loogikaventiil).

Näide

---TEGEMISEL---

Kontroller NOR (mitte või)

NOR on "Not OR" (mitte või) kontroller ehk kontrolleri OR vastand. Kontroller NOR käivitab täitured ainult siis, kui ükski sensoritest pole aktiveerunud.

Näide

---TEGEMISEL---

Kontroller XNOR (välistav mitte või)

XNOR on "Exclusive Not OR" (välistav mitte või) kontroller. Kõlab segaselt, ent tegemist on lihtsalt kontrolleri XOR pöördversiooniga. XNOR käivitab käitured, kui vähem või rohkem kui üks sensor on aktiveeritud.

Näide

---TEGEMISEL---

Avaldise kontroller (*Expression*)

See arvutab kasutaja koostatud loogilise avaldise tulemuse (mille vastuseks on kas tõene või väär, 1 või 0).

Antud juhul kontrollib avaldise kontroller, et sensori signaal nimetusega "sensor" on tõene ja omaduse nimega "prop" väärtus on 1.

Vaata [siit](#).

Pythoni kontroller

Kontroller Python kontrollib sisendit, kasutades selleks programmeeritud skripti ehk siis Pythoni skripti või mõnda muud Pythoni koodi sisaldavat faili. Pythoni kontrolleritel on kaks režiimi: Script (skript) ja Module (moodul). Mõlemad saab kirjutada tekstiredaktoris ja talletada .blend-faili; kasutada võib ka väliseid skriptifaile.

Lisateavet Pythoni kasutamise kohta mängumootoris loe [siit](#).

Pythoni liides mängumootori jaoks on saadaval [siin](#).

Skriptirežiim (*Script Mode*)

Režiimis Script on kontroller skriptiga seotud. Kogu skript käiakse läbi, enne kui loogika käivitub. Kõik skriptis toimuv leiab aset ühes kaadris. Kui samu väärtusi või omadusi muudetakse korduvalt, jääb mängus näha ainult viimane. Näiteks kui objekti asukohta muudetakse kõigepealt positsiooni (100.0,100.0,100.0) ja hiljem muudetakse samas skriptis positsiooni (0.0,0.0,0.0), siis ainult positsioon (0.0,0.0,0.0) jääb mängijale nähtavaks, kuna see kõik leiab aset ühes kaadris.

Moodulirežiim (*Module Mode*)

Pythoni mooduli kontroller lisati Blenderisse versioonis 2.49. Vali kontrolleri Python rippmenüüst skripti (Script) asemel lihtsalt moodul (Module). Siis määratle mooduli otstarve ja pöördu selle poole läbi kontrolleri. Selle asemel et kirjutada skripti tekstiväljale "myScript.py", kirjuta mooduli tekstiväljale "myModule.myFunc". Iga kord kui kontrolleri poole pöördutakse, käivitatakse see funktsioon. Muud funktsioonid ja muutujad väljaspool funktsiooni myFunc käivitatakse ainult üks kord. See on praktiline koodi optimeerimiseks, kui soovid väärtustada muutujad ainult üks kord ja kasutada neid millalgi hiljem.

Pythoni mooduli kontroller toetab lõputult väärtusi, mis tähedab, et automaatselt toetatakse ka Pythoni (tarkvara)pakette. Samaselt nagu "myModule.myFunc" võid teha ka "myPackage.myModule.myFunc". Ka alampaketid (*nested packages*) töötavad, nagu ka meetodi väljakutsed klassist (calls on class) nagu: "myPackage.myModule.myInstance.myMethod". Pythoni kontroller antakse Pythoni funktsioonile ette nagu ühte argumenti kasutavate funktsioonide vastav argument.

See võimaldab skripte käigu pealt muuta. Lisateavet Pythoni mooduli kontrolleri kohta leiad:

*Foorumiteema <http://blenderartists.org/forum/showthread.php?t=156672>

*Video http://download.blender.org/apricot/live_bge_edit.ogv

Mängu avaldised (*Expressions*)

Kontroller võib arvutada välja loogilisi avaldisi, mille alusel saata ühendatud käituritele positiivseid või negatiivseid impulsse:

- kui avaldise tulemus on **tõene**, saadab kontroller ühendatud käituritele positiivse impulsi.
- kui avaldise tulemus on **väär**, saadab kontroller ühendatud käituritele negatiivse impulsi.

Muutujad

Saad kasutada:

- sensorite nimesid**,
- omadusi**: määra mõnele objektile mängu omadus ja kasuta seda mõne kontrolleri avaldises.

Need ei tohi sisaldada tühikuid.

Toimingud

Matemaatilised tehted

Tehted: *, /, +, -

Tulemus: arv

Näited: 3 + 2, 35 / 5

Loogikatehted

- Võrdlustehted: <, >, >=, <=, ==, !=
- Kahendtehted: AND, OR, NOT

Tulemus: tõene (*True*) või väär (*False*).

Näited: 3 > 2 (tõene), 1 AND 0 (väär)

Tingimuslaused (if)

Kasutamine:

```
if( avaldis, impulss_kui_avaldis_tõene, impulss_kui_avaldis_väär )
```

Kui kontrolleris on **avaldis** avaldise tulemus tõene:

- kui **impulss_kui_avaldis_tõene** on väärtusega *True*, saadab kontroller ühendatud käituritele positiivse impulsi.
- kui **impulss_kui_avaldis_tõene** on väärtusega *False*, saadab kontroller ühendatud käituritele negatiivse impulsi.

Kui kontrolleris on **avaldis** avaldise tulemus väär:

- kui **impulss_kui_avaldis_väär** on väärtusega *True*, saadab kontroller ühendatud käituritele positiivse impulsi.
- kui **impulss_kui_avaldis_väär** on väärtusega *False*, saadab kontroller ühendatud käituritele negatiivse impulsi.

Näited

Objektil on omadus **sendid**, mille väärtus on 30:

```
sendid > 20
```

tulemus on tõene (kontroller saadab ühendatud käituritele positiivse impulsi).

Objektil on:

- sensor nimega **Voti_Sees** väärtusega *True*,
- omadus nimega **Kytus** väärtusega *False*,

```
Voti_Sees AND Kytus
```

tulemus on väär (kontroller saadab ühendatud käituritele negatiivse impulsi).

See on sama kui:

```
if (Voti_Sees AND Kytus, True, False)
```

Selle asemel võid kirjutada:

```
if (Voti_Sees AND Kytus, False, True)
```

mis saadaks positiivse impulsi, kui **Voti_Sees** **AND** **Kytus** annab vastuseks väära.

Võid samuti kirjutada:

```
if ((Voti_Sees AND Kytus) OR (sendid > 20), True, False)
```

Selle avaldise tulemus oleks tõene ning sellisel juhul saadaks kontroller ühendatud käituritele positiivse impulsi.

Käituriid (Actuators)

Käituriid teostavad tegevusi nagu objektide loomine ja liigutamine ning heli mängimine. Käituriid teostavad oma funktsiooni, kui nad saavad positiivse impulsi mõnelt (või mitmelt) oma kontrollerilt.

Kasutatavad käituriid on järgnevad:

Liikumine (Motion)	Paneb objekti liikuma ja/või pöörlema. Erinevad võimalused "telepordist" kuni objektide füüsiliselt pöörlema lükkamiseni.
Kuju muutja (Shape Action)	Teostab animatsioone võtmevormidesse talletatud kujude ja kujumuutjatega (Shape Actions).
Tegevus (Action)	Juhib skeleti tegevusi. See on nähtav vaid siis, kui skelett on selekteeritud.
Piiraja (Constraint)	Piirajaid kasutatakse objekti asukoha, kauguse või pöörde limiteerimiseks. Need on vajalikud objektide füüsika kontrollimiseks mängus.
F-kõver (F-Curve)	Juhib animatsiooni kõveraid, mille abil saab objekte liigutada, pöörata, mõõtkava teisendada, värvi muuta või teha muid taolisi tegevusi.
Kaamera (Camera)	Võimaldab objekte sujuvalt jälitada. Mõeldud peamiselt kaameraobjektidele, ent kasutatav iga objekti puhul.
Heli (Sound)	Kasutatakse mängus helide esitamiseks.
Omadused (Property)	Muudab objekti omadusi neid määrates, lisades või kopeerides.
Objekti muutmine (Edit Object)	Muudab objekti võret. Lisab objekte, lõhub neid ja võib ka objekti võret muuta (ja varsti ka pörkevõret (<i>collision mesh</i>) uuesti luua).
Stseen (Scene)	Haldab stseene sinu .blend-failis, mida saab kasutada mängutasemetel, kasutajaliidese või taustana
Juhuslik (Random)	Loob juhuslikke väärtusi, mida saab omadustesse salvestada.
Teade (Message)	Saadab teateid, mida teised objektid saavad vastu võtta (ja mis käivitavad vastava objekti).
Mäng (Game)	Haldab tervet mängu ja võimaldab tegevusi nagu uuesti alustamine, sulgemine, laadimine ja salvestamine.
Nähtavus (Visibility)	Muudab objekti nähtavust.
2D-filtrid (2D Filters)	Filtrid eriefektidele nagu seepia värvid või hägu.
Allutamine (Parent)	Võimaldab määrata objektile ülemobjekti või eemaldada selle.
Olek (State)	Muudab objekti olekuid.

Tegevuse käituri (*Action*)

Tegevuse käituri on nähtav ainult siis, kui valitud on mõni skelett, sest tegevused salvestatakse skeletti.

Action Playback Type (tegevuse tagasimängimise tüüp)

- *Play* (mängi)
- *Ping Pong*
- *Flipper* (edasi-tagasi)
- *Loop End* (korda lõppu)
- *Loop Start* (korda algust)
- *Property* (omadus)

Action (tegevus)

Vali kasutatav tegevus

Continue (jätkka)

Taasta sisse/välja lülitades viimane kaader, vastasel juhul mängitakse iga kord algusest

Start Frame (alguskaader)

Määra tegevuse alguskaader

End Frame (lõppkaader)

Määra tegevuse lõppkaader

Frame Blend In (sisseulandamise kaadrid)

Tegevuse sulandamise (*motion blending*) kaadrite arv

Priority (prioriteet)

Käivitamise prioriteet - madalamad numbrid kirjutavad üle kõrgemate numbritega tegevused. Kui korraga toimub kaks või enam tegevust, peavad üle kirjutavad kanalid olema pinus eespool

Frame Property (kaadri omadus)

Määra tegevuse hetke kaader selle omaduse väärtuseks

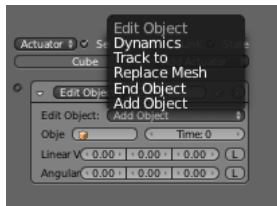
Property (omadus)

Kasuta tegevuse asukoha määramiseks seda omadust. Töötab ainult omaduse (*Property*) mahamängimise tüübiga.

Objekti muutmise käitur (*Edit Object*)

Objekti muutmise (Edit Object) käitur võimaldab kasutajal mängusiseselt objekti seadeid muuta

[[File:2.5_Game_Engine_Actuator_Edit_Object.png|thumb|right|271px|Edit Object käitur]]



Edit Object käituri seaded

- **Edit Object** (muuda objekti)
- **Dynamics** (dünaamika)
- **Track To** (jälita)
- **Replace Mesh** (asenda võre)
- **End Object** (lõpeta objekt)
- **Add Object** (lisa objekt)

Dünaamika (*Dynamics*)

Set Mass (määra mass)

Määrab füüsika jaoks parajasti valitud objekti massi.

Disable Rigid Body (lülita jäik keha välja)

Lülitab välja objekti jäiga keha staatuse, st kokkupõrgete arvutamise.

Enable Rigid Body (lülita jäik keha sisse)

Lülitab sisse objekti jäiga keha staatuse, st kokkupõrgete arvutamise.

Suspend Dynamics (Peata dünaamika)

Peatab objekti liikumise (objekti kiiruse).

Restore Dynamics (Taasta dünaamika)

Taastab objekti liikumise (objekti kiiruse).

Jälita (*Track To*)

Object (objekt)

Objekt, mida jälitada.

Time (aeg)

Lülita sisse sujuv jälitus (sisaldab viivitust).

3D

Luba 3D-jälitust (jälitatakse X-, Y-, Z-telgedel).

Asenda võre (*Replace Mesh*)

Mesh (võre)

Uue võre nimi.

Gfx (graafika)

Asenda nähtav võre.

Phys (füüsika)

Asenda füüsika võre (mitte kompleksed vormid).

Lõpeta objekt (*End Object*)

Lõpetab antud objekti (Märkus: vigadeotsija näitab konsoolis Zombie Objekti viga)

Lisa objekt (*Add Object*)

Object (objekt)

Objekt, mida jälitada.

Time (aeg)

Aeg (loogikaarvutuse sammudes) kuni objekti 'surmani'.

Linear Velocity (lineaarne kiirus)

Lisatud objekti esmane lineaarne kiirus.

Angular Velocity (nurkkiirus)

Lisatud objekti esmane nurkkiirus.

2D-filtri käitur (2D Filter)

2D-filtrid (2D Filter) on pildi filtreerimise käitured (*Actuator*), mis rakenduvad objektide lõpprenderdusel. Tänuavaldused jms FPS_Template'i eest.

Liikumishägusti

Liikumishägusti Motion Blur on 2D-filter, mis vajab eelnevat renderdusinformatsiooni objektide liikumise efekti tekitamiseks. Allpool näed liikumishägusti filtrit (Motion Blur) Blenderi aknas koos loogikaklotsidega:



2D-filtrid: Liikumishägusti (Motion Blur).



2D-filtrid: Mänguloogika.

Filtri sisselülitamiseks:

1. Ühenda vastav(ad) sensor(id) ja kontroller(id).
2. Lisa 2D-filtri käitur (2D Filter).
3. Vali rippmenüüst liikumishägusti (Motion Blur).
4. Määra liikumishägusti väärtus (Value).

Ja filtri väljalülitamiseks:

1. Ühenda vastav(ad) sensor(id) ja kontroller(id).
2. Lisa 2D-filtri käitur (2D Filter).
3. Vali liikumishägusti (Motion Blur).
4. Lülita nupp Enable (toimiv) välja.

Liikumishägu filtrit saab sisse lülitada ka Pythoni kontrolleri abil:

```
bge import render
render.enableMotionBlur(0.85)
```

Ja välja lülitada:

```
bge import render
render.disableMotionBlur()
```

Märkus

Su graafika riistvara ja OpenGL-i draiver peavad toetama akumulatsiooni puhvrit (*glAccum*-funktsiooni).

Sisseehitatud 2D-filtrid

Kõigil 2D-filtri käituri nähtavatel 2D-filtritel on ühtne arhitektuur. Kõik sisseehitatud filtrid kasutavad lõpliku renderdusvaate esitamiseks fragmentvarjutust, seega peab su riistvara toetama varjutusalgoritme.



2D-filtrid: Liikumishägusti (Motion Blur).



2D-filtrid: Seepia (Sepia).



2D-filtrid: Sobel.

Blur (hägu), Sharpen (teravdus), Dilation (paisutus), Erosion (erosioon), Laplacian, Sobel, Prewitt, Gray Scale (halltoonid), Sepia (seepia) ja Invert (negatiiv) on sisseehitatud filtrid. Neid filtreid saab teatud käikudes muuta kättesaadavaks.

Filtri kasutamiseks:

1. Tekita sobiv(ad) sensor(id) ja kontroller(id):
2. Loo käitur 2D Filter.
3. Vali oma filter, näiteks Blur (hägu).
4. Määra käigu number, kuhu filter rakendatakse.

Filtri eemaldamiseks konkreetses käigus:

1. Tekita sobiv(ad) sensor(id) ja kontroller(id).
2. Loo käitur 2D Filter.
3. Vali Remove Filter (eemalda filter).
4. Määra käigu number, kust soovid filtri eemaldada.

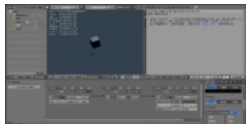
Filtri blokeerimiseks konkreetses käigus:

1. Tekita sobiv(ad) sensor(id) ja kontroller(id).
2. Loo käitur 2D Filter.
3. Vali Disable Filter (lülita filter välja).
4. Määra käigu number, kus soovid filtri blokeerida.

Filtri konkreetses käigus sisse lülitamiseks:

1. Tekita sobiv(ad) sensor(id) ja kontroller(id).
2. Tekita käitur 2D Filter.
3. Vali Enable Filter (lülita filter sisse).
4. Määra käigu number, kus soovid filtri sisse lülitada.

Kasutaja loodud filtrid



2D-filtrid: oma filter
(Custom Filter).

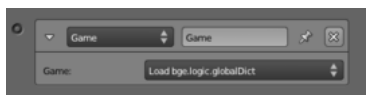
Omalooodud filtrid võimaldavad sul GLSL-i keelt kasutades määrata enda 2D-filtri. Nende kasutusviis on sama nagu sisseehitatud filtritel, kuid sa pead 2D-filtri käituri valima omalooodud filtri (Custom Filter), siis kirjutama tekstiredaktoris varjutusprogrammi ja kirjutama varjutamisalgoritmi nime käituri sisse.

Sinise seepia näide:

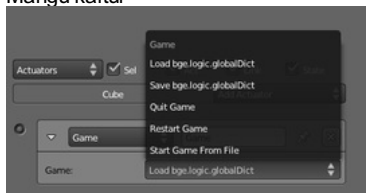
```
uniform sampler2D bgl_RenderedTexture;
void main(void)
{
    vec4 texcolor = texture2D(bgl_RenderedTexture, gl_TexCoord[0].st);
    float gray = dot(texcolor.rgb, vec3(0.299, 0.587, 0.114));
    gl_FragColor = vec4(gray * vec3(0.8, 1.0, 1.2), texcolor.a);
}
```

Mängu käitür (*Game*)

Mängu käitür võimaldab kasutajal teostada mänguspetsiifilisi funktsioone nagu mängu taaskäivitamine, mängu sulgemine ja mängu laadimine.



Mängu käitür



Mäng

Tüüp (*Type*)

Load bge.logic.globalDict

Lae bge.logic.globalDict failist .bgeconf.

Save bge.logic.globalDict

Salvesta bge.logic.globalDict faili .bgeconf.

Quit Game (sulge mäng)

Niipea kui käitür on käivitatud, väljub blenderimängija programmist.

Restart Game (taaskäivita mäng)

Niipea kui täitur on käivitatud, taaskäivitab blenderimängija mängu (laeb uuesti failist).

Start Game From File (lae mäng failist)

Niipea kui käitür on käivitatud, käivitab blenderimängija määratud asukohas oleva .blend-faili.

File (fail)

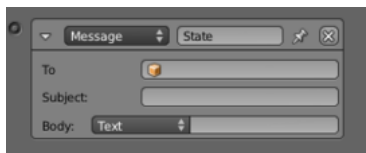
Laetava .blend-faili asukoht.

Märkused kasutamise kohta

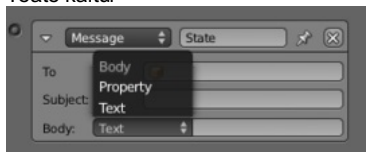
Kui sa kasutad klaviatuurisensorit klahvi Esc haagina, siis mängu sulgemise käituri äpardumise korral (näiteks vea tõttu Pythoni failis) on mängu võimatu sulgeda. Andmeid saab taastada quit.blend-failist. File » Recover Last Session (fail->taasta eelmine sessioon)

Teate käitur (*Message*)

Teate käitur võimaldab kasutajal saata andmeid üle stseeni ja isegi stseenide vahel.



Teate käitur



Teate käituri sätted

To (saaja)

Objekt, millele teade edastatakse. Jäta tühjaks, kui saadad kõigile (või kui saadad teise stseeni).

Subject (teema)

Sõnumi teema. Sobiv teatud tüüpi info edastamiseks, näiteks mängu lõpu sõnumi (*end-game*) saatmine sõnumisensorile, mis ootab sõnumit *end game* ja saadab selle edasi kontrollerile AND, mis omakorda saadab selle edasi mängu lõpetamise käiturile *Quit Game*

Body (sisu)

Sõnumi sisu (ainult Pythoni poolt loetav).

Text (tekst)

Kasutaja koostatud sisutekst.

Property (väärtus)

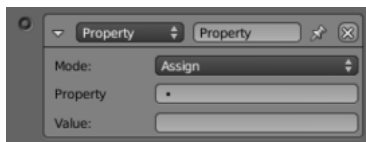
Kasutaja määratud väärtus, näiteks mängu skoor.

Märkus kasutamise kohta

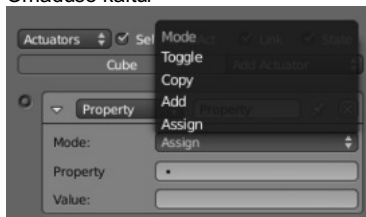
Sõnumikäiturit võib kasutada andmete edastamiseks, näiteks skooride saatmiseks teistele objektidele või isegi stseenide vahel (selleks võib kasutada ka `bge.logic.globalDict`).

Omaduse käituri (*Property*)

Omaduse käiturit kasutades saad muuta etteantud omaduse väärtust, kui käituri ise on aktiveeritud.



Omaduse käituri tüübid



Omaduse käituri tüübid

Prop (omadus)

Omadus, mida käituri hakkab muutma

Value (väärtus)

Väärtus, mis omadusele antakse

Type (tüüp)

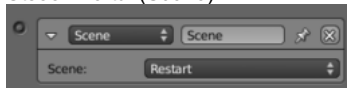
- **Assign (määra)**: Prop-omadusele antakse käituri määratud väärtus (Value), niipea kui käituri on käivitatud
- **Add (liida)**: Liidab käituri määratud väärtuse (Value) omaduse (Prop) väärtusele, niipea kui käituri on käivitatud (väärtuse vähendamiseks sisesta negatiivne väärtus). Kahendmuutuja (Bool) puhul loetakse tõeseks nullist erinev (ka negatiivne) väärtus.
- **Copy (kopeeri)**: kopeerib käituri käivitamisel mõne teise objekti omaduse väärtuse käituri omaniku omadusele.
- **Toggle (vaheta)**: lülitab nulli üheks ja iga muu numbri peale nulli nulliks, niipea kui käituri on käivitunud. Kasulik sisse/välja lülitite puhul.

Näide

Sul on tegelaskuju, tal on omadus nimega *hp* (hit points ehk elupunktid) otsustamiseks, millal tema vigastused on nii suured, et ta sureb. *hp* on täisarv algväärtusega 100.

Seadista kaks põrkesensorit (Collision), üks vaenlase kuulidele ja teine tervise taastamiseks. Esimene on ühendatud (läbi AND kontrolleri) omaduse liitmise käituri Add Property, mille omaduseks on *hp* ja väärtuseks -10. Iga kord kui mängijat tabab vaenlase kuul, kaotab ta 10 hp. Teine sensor on ühendatud (läbi AND kontrolleri) teise omaduse lisamise käituri (Add Property) ja selle väärtuseks on 50. Seega iga kord kui mängija põrkub tervendava esemega, tõuseb tema *hp* 50 võrra. Järgmisena seadista omaduse sensor (Property) vahemiku jaoks, mis on suurem kui 100. See on ühendatud (läbi AND kontrolleri) omaduse liitmise käituri (Assign Property), mille väärtus on 100. Seega, kui mängija *hp* tõuseb üle 100, muudetakse see 100-ks.

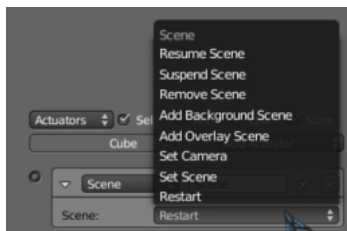
Stseeni käitür (Scene)



Stseeni käitür

Stseeni käitür (Scene) haldab sinu .blend-failis olevaid stseene, mida saab kasutada mängutasemetel, kasutajaliidese või taustana.

Sellel käitüri on kaheksa režiimi:



Stseeni käituri seaded

Restart (alusta uuesti)

Alusta stseeni uuesti. Kõik stseenis olev läheb tagasi algseesse seadistusse

Set Scene (määra stseen)

Vahetab olemasoleva stseeni määratud stseeni vastu

Set Camera (määra kaamera)

Määrab kasutatava kaamera

Add OverlayScene (lisa esiplaani stseen)

See lisab teise stseeni, mis kuvatakse parajasti avatud stseeni peale. See on kasulik kasutajaliidese puhul, sest tervise riba, laskemoona mõõdiku või kiiruse mõõdiku näitamine pealekuvatavas stseenis muudab nad alati nähtavaks

Add BackgroundScene (lisa taustastseen)

See on esiplaani stseeni vastand ja kuvatakse parajasti avatud stseeni taustaks

Remove Scene (eemalda stseen)

Eemaldab stseeni!

Suspend Scene (peata stseen)

Peatab stseeni mängimise

Resume Scene (jätkka stseeni)

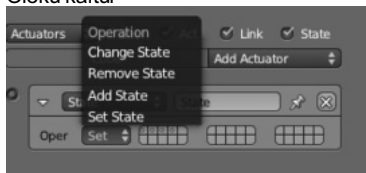
Jätkab peatatud stseeni mängimist.

Oleku käitur (*State*)

Oleku käitur võimaldab kasutajal luua keerulist loogikat, säilitades samal ajal selge kasutajaliidese. Tal on võimalikud erinevad erinevad olekud ja ta teostab vastavalt nendele erinevaid toiminguid.



Oleku käitur



Oleku käituri seaded

Toiming (*Operation*)

Change State (muuda olekut)

Muudab oleku praegusest määratud olekusse.

Remove State (eemalda olek)

Eemaldab määratud olekud aktiivsete olekute seast (deaktiveerib nad).

Add State (lisa olek)

Lisab määratud olekud aktiivsete olekute sekka (aktiveerib nad).

Set State (määra olek)

Liigub praegusest olekust määratud olekusse, deaktiveerides teised lisatud olekud.

Märkus kasutamise kohta

Oleku käituri abil võid luua loogikakihte, ilma et oleks vaja sadu omadusi. Kasuta seda arukalt ja sul on sellest palju kasu. Probleemid saab tavaliselt lahendada Pythoni abil.

Nähtavuse käitur (*Visibility*)

Nähtavuse käituri saab kasutada mängimise ajal objektide nähtavust muuta.



Nähtavuse käitur

Visible (nähtav)

Lülitab nähtavuse sisse ja välja

Occlusion (peitmine)

Peitmise sisse ja välja lülitamiseks tee kasti linnuke. See tuleb Füüsika alt kõigepealt käivitada.

Children (alamad)

Rekursiivseks peitmiseks tee kasti linnuke, see määrab nähtavuse/peitmise kõigile alamaobjektidele, alamate alamatele jne.

Märkus kasutamise kohta

Nähtavuse käituri kasutamine hoiab kokku rasteriseerimise (*Rasterizer*), kuid mitte füüsika arvelt ning on seega detailsustaseme (*LOD, Level of Detail*) määramiseks piiratud kasulikkusega. Detailsustaseme muutmiseks uuri võrede asendamist, kuid pea meeles, et selleks vajaminev loogika võib saavutatavat efekti vähendada.

Kaamera Blenderi mängumootoris

Mängumootoris saab kaamerat kasutada vaikeomadustega (kuidas see liigub) või siis allutada selle objektile või objekti tipule või [kaamera käituri](#) abil objekti jälitada.

Märkus

Iga objekt võib muutuda kaameraks (loe altpoolt) ning kaameraga objekti jälitamise puhul võib kasutada ükskõik millist objekti (näiteks tühiobjekti).

Kaamera kasutamiseks pead mängumootori käivitama kaamera vaateaknast (0 NumPad) ning kaamera moonutuse vältimiseks pead alati vaadet senikaua suurendama, kuni kaamera objekt täidab terve vaateakna.

Vaikekaamera

See on vaade maailmale, mille puhul kaamera ei liigu.

Kaamera allutamine objektile

Kaamera liigub koos objektiga. Vali kõigepealt kaamera ja seejärel objekt. Seejärel vajuta CtrlP → Make Parent (muuda ülemobjektiks).

Pane tähele, et kui objektile on määratud pööre, on see pööre ka kaameral. Selle vältimiseks vali "tipule allutamine" (*Parent to Vertex*, loe altpoolt).

Tipule allutamine

Kõige lihtsam moodus selle saavutamiseks on valida objekt ning minna klahvi ⇄ Tab abil muutmisrežiimi Edit. Vali nüüd tipp ja mine klahvi ⇄ Tab abil tagasi objektirežiimi Object.

Järgmisena, nii et ühtegi objekti pole valitud, vali kaamera ning seejärel klahvi ⇄ Shift all hoides ka objekt. Mine klahviga ⇄ Tab muutmisrežiimi Edit, vajuta CtrlP ja vali tipule allutamine (Make vertex parent).

Nüüd järgneb kaamera objektile ning säilitab oma pööre isegi siis, kui objekt pöörab.

Objekt kui kaamera

Iga objekt võib olla ka kaamera koos kõigi sellele objektile määratud omadustega.

Objekti kaameraks muutmiseks vali see objektirežiimis Object ning vajuta klahvikombinatsiooni Ctrl0 NumPad.

Selle tühistamiseks vali kaamera ja vajuta uuesti Ctrl0 NumPad.

Kaamera läätse nihutamine

Blenderi kasutajaliidesega saab kaamera vaadet vaate X/Y-tasapinnal liigutada. See on võrreldav videoprojektorite läätse nihutamisega, mis tavaliselt liigutavad pilti mööda Y-telge üles (see on mõeldud selleks, et kui sa asetad projektori laua peale, ei näitaks see poolt pildist laua pinnale).

Kahjuks ei arvesta mängumootor selle parameetriga.

Projektsiooni muutmiseks saame me kaamera projektsioonimaatriksit (*camera projection matrix*) otse Pythonis muuta.

```
import bge
scene = bge.logic.getCurrentScene()
cam = scene.active_camera
# leia projektsiooni maatriks
camatrix = cam.projection_matrix
# muuda kaamera projektsiooni maatriks, muutes kolmanda rea x-i ja y-i liikmeid, et need omandaksid renderdatava ala nihke
camatrix[2][0] = 2*shiftx
camatrix[2][1] = 2*shifty
cam.projection_matrix = camatrix
```

shiftx ja shifty on siinkohal vaateala ühikutes ning seega näiteks soovides nihutada vaadet poole ekraani võrra ülespoole, määratakse shifty väärtuseks 0.5.

Loe lisaks

- [Kalasilm-kuppelkaamera](#).

Stereokaamera (*Stereo Camera*)

Stereokaamera abil saad renderdada pilte, mis näivad läbi spetsiaalsete prillide vaadatuna kolmemõõtmelistena. Selle saavutamiseks renderdatakse kaks eraldi pilti kaameratega, mis asuvad teineteise suhtes kerges nihkes, simuleerides seda, kuidas meie endi silmad näevad. Stereopilti vaadates näeb üks silm ainult ühte ja teine teist pilti. Meie aju suudab need kokku ühendada, mille tulemusena näib, nagu vaataksime me lameda pildi asemel kolmemõõtmelist objekti.

Stereo seaded

Stereo Mode (stereorežiim)

Määra kasutatava stereokaamera tüüp. Need on lahti seletatud allpool.

Eye Separation (silmadevaheline kaugus)

See on äärmiselt oluline. See määrab, kui kaugel üksteisest need kaks pilti on ja kui kolmemõõtmelistena seetõttu näivad. Liiga kõrge väärtus võib põhjustada peavalu ja silmade väsimist.

Stereorežiimid

Nelikpuhver (*Quad Buffer*)

Üleval-all (*Above-Below*)

Ülerealaotus (*Interlaced*)

Anaglüüf (*Anaglyph*)

Kõrvuti (*Side by Side*)

Vertikaalne ülerealaotus (*Vinterlaced*)

Mängumootori kuplirežiim (*Dome Mode*)

Selle abil saavad kunstnikud visualiseerida interaktiivseid projekte panoraamkinos. Eesmärgiga muuta see tööriist laiendatavaks, toetab Blender täiskuplit (*Full dome*), lõigatud kupleid (*Truncated dome*, eest ja tagant), planetaatriume (*Planetarium*) ja kerajate peeglitega kupleid.

See võimalus lisandus Blenderi versioonis 2.49. Kuplirežiim kasutab mitmekäigulist (*multipass*) tekstuurialgoritmi, mille arendas välja Paul Bourke ning mille lisas Blenderisse Dalai Felinto tänu **SAT** (Society for Arts and Technology, kunstide ja tehnoloogia selts) toetusele **SAT Metalab** süvauuringute programmi raames. [\[1\]](#) Lühidalt tähendab see stseeni neljakordset renderdamist ning saadud piltide asetamist võrele, mis on disainitud sellisena, et tulemus näiks läbi paralleelprojektsioonis kaamera vaadatuna nagu kalasilm-projektsioon.

Märkus

Pea meeles, et Blenderit tuleb kasutada täisekraani režiimis, et projektor annaks maksimaalse efekti.

Selle saavutamiseks käivita Blender käsurea argumendiga -W. Et saada lahti Blenderi menüüst, proovi kõik aknad (nupud, 3D-vaade, tekst jne) üheks liita. Kui sa lihtsalt programmi maksimeerid (Ctrl+Up), ei kata mäng tervet ekraani (ülemine ribamenüü võtab enda alla 20 pikslit).

Kuppelkaamera seeded

Dome Type (kupli tüüp)

Selle menüü kaudu saad valida kasutatava kuppelkaamera tüübi. Variandid on koos sätetega toodud allpool.

- [Fisheye Dome](#) (kalasilm)
- [Front-Truncated Dome](#) (eest lõigatud kuppel)
- [Rear-Truncated Dome](#) (tagant lõigatud kuppel)
- [Cube Map](#) (kuuplaotus)
- [Full Spherical Panoramic](#) (täiskera panoraam)

Kasutatavad kaamerasätted muutuvad vastavalt valitud kupli tüübile:

Resolution (lahutus)

Määrab puhvri lahtuse. Väärtuse vähendamine suurendab renderduskiirust, kuid vähendab kvaliteeti.

Tessellation (tessellatsioon)

Vaikeväärtus on 4. See on võre tessellatsiooni määr. (Kuuplaotuse režiimis seda kasutada ei saa.)

Angle (nurk)

Määrab kupli vaatevälja kraadides, 90 kuni 250. (Kasutatav kalasilma ja lõigatud režiimides.)

Tilt (kalle)

Määrab kaamera kalde horisontaalse telje suhtes. (Kasutatav kalasilma ja lõigatud režiimides.)

[Warp Data](#) (väände andmed)

Kasuta spetsiaalselt loodud võre väände andmefaili.

Kalasilma režiim (*Fisheye*)

Paralleelne kalasilma vaade, nurgaga on 90° kuni 250° kraadi.

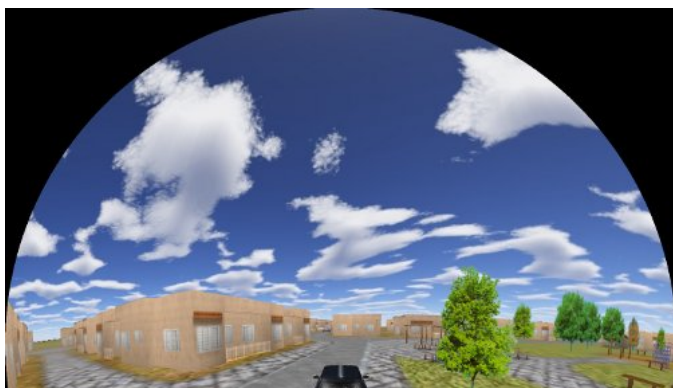
- Nurkade 90° kuni 180° puhul kasutatakse nelja renderdust.
- Nurkade 181° kuni 250° puhul viite renderdust.



Eest lõigatud kuppel (*Front-Truncated Dome*)

See on mõeldud lõigatud kuplite jaoks: selles režiimis joondatakse kalasilma pilt akna ülemise äärega nii, et see puudutaks külgi.

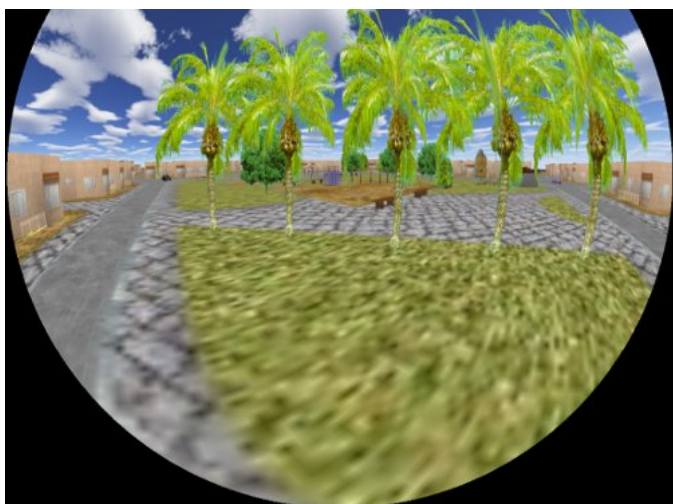
- Vaateväli on 90° kuni 250° kraadi.
- Nurkade 90° kuni 180° puhul kasutatakse nelja renderdust.
- Nurkade 181° kuni 250° puhul viite renderdust.



Tagant lõigatud kuppel (*Rear-Truncated Dome*)

See on mõeldud lõigatud kuplite jaoks: selles režiimis joondatakse kalasilma pilt akna alumise äärega nii, et see puudutaks külgi.

- Vaateväli on 90° kuni 250° kraadi.
- Nurkade 90° kuni 180° puhul kasutatakse nelja renderdust.
- Nurkade 181° kuni 250° puhul viite renderdust.



Kuuplaotuse režiim (Cube Map)

Kuuplaotuse režiimi saab kasutada eelanimeeritud kuuplaotusega (*CubeMap*) piltide loomiseks.

- Selle puhul kasutatakse kuut renderdust. Nende piltide järjekord järgib Blenderi sisemist keskkonnalaotuse EnvMap failiformaati:
 - esimene rida: parem, tagumine, vasak
 - teine rida: alumine, ülemine, esikül



Kerapanoraam (*Spherical Panoramic*)

Täiskera panoraamrežiim.

- Selle puhul kasutatakse kuut kaamerat.
- Alumine ja ülemine saavad täpseks siis, kui välja **Definition** väärtus on 5 või enam.



Väändevõre andmed (*Warp Data Mesh*)

Mitmed projektsioonikeskkonnad nõuavad pilte, mis ei oleks lihtsalt perspektiivsed projektsioonid (mis on tavapärased lamedate ekraanide puhul). Näiteks geomeetria sobitamine silinderjate ekraanide ning mõnede uute planetaariumikuplite või virtuaalse reaalsuse jaoks mõeldud seisvate kuplite puhul.

Võre formaadi kohta loe täpsemalt [Paul Bourke'i artiklist](#).



Selliste piltide loomiseks kasutame me erilist failiformaati.

Faili näidis::

```
mode
width height
n0_x n0_y n0_u n0_v n0_i
n1_x n1_y n1_u n1_v n1_i
```

```
n2_x n1_y n2_u n2_v n2_i  
n3_x n3_y n3_u n3_v n3_i  
(...)
```

Esimene rida tähistab pildi tüüpi, millele võre määratakse: **2 = nelinurkne**, **1 = ümmargune**. Teises reas on võre mõõtmised pikslites. Ülejäänud read on võre sõlmed.

Iga rida koosneb **x y u v i** väärtustest. (x,y) on normaliseeritud ekraani koordinaadid. (u,v) on tekstuurikoordinaadid. i on kordistav intensiivsusefaktor (*multiplicative intensity factor*).

x sõltub ekraani laiuse/kõrguse suhtest. y väärtus on -1 kuni 1. u ja v väärtused 0 kuni 1. i väärtus on 0 kuni 1 - kui selle väärtus on negatiivne, siis seda võre sõlme ei kuvata.

- Sa pead selle faili looma ja lisama tekstiredaktorisse, et saaksid seda oma väändvõre andmefailina kasutada.
- Ava tekstiredaktor (*Window Types/Text Editor*).
- Ava oma võre andmete fail (st myDome.data) tekstiredaktoris (menüü *Text/Open* (tekst->ava) või kiirklahv Alt O).
- Mine mängukaadri seadetes (*Game Framing Settings*) - *Window Types/Buttons Window/Scene Page* (akna tüübid/nuppude aken/stseeni leht) või kiirvalik F10
- Lülita sisse kuplirežiim (*Dome Mode*).
- Kirjuta väljale *Warp Data* (väände andmed) faili nimi (nt myDome.data).

[Paul Bourke'i Warpplayeri](#) tarkvarapaketi osana on saadaval võrelaotaja (*meshmapper*) nimeline interaktiivne tööriist, millega saad ise oma väändvõresid luua (nõuab täisversiooni olemasolu).

Näidisfailid

[Näidis: peegelkuppel 4×3](#), [Näidis: lõigatud kuppel 4×3](#), [Näidis: täisekraani fail 4×3](#), [Näidis: täispuhvri fail 4×3](#).

Märkus

Tähtis: vaateava arvutamiseks kasutatakse lõuendi laiuse ja kõrguse suhet. Seetõttu nõuavad erinevad ekraanisuurused erinevaid väändvõre faile. Et külgede suhe sobiks sinu projektorile, pead kasutama Blenderit täisekraani režiimis.

Mängufüüsika

Mängufüüsika seaded asuvad maailma paneelis (*World*). Need määravad mängumootori stseenis toimivad füüsikaseadused ja gravitatsiooni. Vastavalt valitud füüsikamootorile liigutab Blender mängumootori füüsikasimulatsioonide ajal tegijaid (Actors) automaatselt allapoole -Z suunas. Kui oled tegijad kohale paigutanud ja nad liiguvad sinu soovide järgi, võid selle väljaarvutatud liikumise eelrenderdada fikseeritud IPO-kõverateks (lisainfot loe [loogika tegijate](#) (*logic actors*) lehelt).

Seaded

Physics Engine (füüsikamootor)

Määrab kasutatava füüsikamootori.

Bullet

Vaikimisi valitud füüsikamootor - seda arendatakse praegu aktiivselt. See käsitseb liikumist ja põrketuvastust (*collision detection*). Asjad, mis põrkuvad, annavad oma liikumisinertsit edasi nendele objektidele, mille vastu nad põrkusid.

None (puudub)

Füüsikamootorit ei kasutata. Gravitatsioon ei mõjuta esemeid ning nad saavad virtuaalses ruumis vabalt ringi lennata. Liikuvad objektid jäävadki liikuma.

Gravity (gravitatsioon)

Selle maailma raskuskiirendus, meetrites ruutsekundi kohta ($\text{m} \cdot \text{s}^{-2}$). Igal objektil, mis on tegutseja, on massi ja suuruse liugur (vaata [objekti füüsika](#) osast).

Võttes arvesse nende väärtusi ja kaadrisagedust (vaata [renderduse](#) osa), arvutab Blender välja, kui suure kiirendusega peaks objekt allapoole langema.

Physics Steps (füüsika sammud)

Max (maksimum)

Määrab maksimaalse füüsikasammude arvu ühes mängu kaadris, kui mängu kiirus on piiratud graafika poolt. Kõrgema väärtuse puhul püsib füüsika reaalajas.

Substeps (alamsammud)

Määrab simulatsiooni alamsammude arvu ühe füüsika ajasammu kohta. Kõrgema väärtuse puhul on füüsika täpsem.

FPS (kaadreid sekundis)

Määrab nominaalse mängu kaadrite arvu sekundis. Füüsika fikseeritud ajasamm = 1/fps ja on tegelikust kaadrisagedusest sõltumatu.

Logic Steps Max (maksimaalne loogikasammude arv)

Määrab maksimaalse loogikasammude arvu ühes mängu kaadris, kui mängu kiirus on piiratud graafika poolt. Kõrgema väärtuse puhul on sünkroniseeritus füüsikaga parem.

Occlusion Culling (peitjaga vaatepiiramine)

Kasuta vaatekoonuse ja peitjaga vaate piiramiseks optimeeritud Bulleti füüsikamootori puud (DBVT).

Resolution (lahutus)

Peitmispuhvri (*occlusion buffer*) suurus pikslites, suurema täpsuse jaoks kasuta kõrgemat väärtust (aeglasem).

Objektide tüübid



Objekti tüüpe

(*Object type*)

menüü

Mängumootoris saavad objektid olla erinevat tüüpi. Füüsika tüübi menüü (Physics Type) kaudu saad sa määrata kõik aktiivse objektiga seotud omadused. Aktiivseid objekte käsitleb füüsikamootor ning nendega tegeldakse rohkem kui tavaliste maailmaobjektidega.

Sätted

Iga objekti füüsika tüüpi saab määrata omaduste paneeli (*Properties*) füüsika alampaneelil (*Physics*).

Valida saad järgmiste objektitüüpide vahel. Nende erilisi omadusi kirjeldatakse vastavatel lehtedel:

- [Peitja \(Occluder\)](#)
- [Põrge puudub \(No Collision\)](#)
- [Sensor](#)
- [Staatiline \(Static\)](#)
- [Dünaamiline \(Dynamic\)](#)
- [Jäik keha \(Rigid Body\)](#)
- [Pehme keha \(Soft Body\)](#)

Põrkepiirid (*Collision Bounds*)

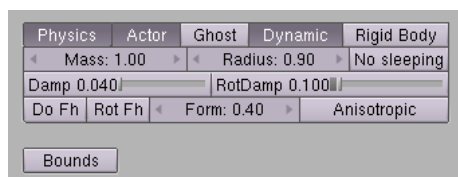
Igal füüsika tüübil on põrkepiiride sätted. Blenderi mängumootor saab objektide kuju käsitleda mitmel erineval viisil. Põhjus, miks mängumootor väldib üliäpseid küljepõhiseid objektide põrkeid, on selles, et see nõuaks väga palju arvutamisvõimsust. Mängimise ajal on sageli kiirem kasutada lihtsat süsteemi nagu kast või kera, kui objekti kuju on nendega piisavalt sarnane.

Piirid arvutatakse objekti keskpunktist, mida tähistab sinu objekti keskel olev (tavaliselt roosa) punkt. Ebatavalised pole ka juhud, kus see keskpunkt on praktilistel põhjustel mõnda loogilisemasse kohta paigutatud. Seda saab teha muutmiskontekstis (Editing - F9), valides seal ühe kolmest keskpunkti liikumise valikust (rohkem infot leiad [objekti keskmete](#) osast).

Kumer kest (Convex Hull) ja staatiline kolmnurkvõre (Triangle Mesh) on mõeldud maailma objektide jaoks: põrandad, seinad, puud jne.

Blenderil puudub üldine moodus seest tühjade objektide käsitlemiseks. Raskesti loodava objekti näitena võiks tuua tassi, millega saaks täringuid raputada.

Objekti füüsika, kui aktiivset maailma ei ole



Objekti füüsika, kui aktiivset maailma ei ole

Kui sul varjutuse kontekstis (*Shading*) (nuppude aken, F5) aktiivne maailm ([World](#)) puudub, näed sa objekti tüübi paneelis (Object type) lihtsalt nuppu nimega füüsika (*Physics*): selle sisselülitamisel saad sa vajutada teistele nuppudele, et kuvataks paneelid nagu pildil *Objekti füüsika, kui aktiivset maailma ei ole*. Sellisel juhul kasutatavaid seadeid [kirjeldatakse siin](#).

Üldjuhul peaksid sa looma maailma, sest see tekitab ka füüsikalise keskkonna.

Staatiline objektitüüp (*Static*)

Füüsikaline objekt, mida ei mõjuta gravitatsioon.

Tegija (*Actor*)

See paneb füüsikamootori sinu objekti olekut jooksvalt ümber arvutama, muutes selle millekski enamaks kui lihtsalt maailma passiivne osa (tavaline sein või põrand). See objekt muutub teistele aktiivsetele objektidele "nähtavaks".

Staatilist objekti tavapärase füüsika (nt gravitatsioon) ei mõjuta, kuid seda nähakse enama kui lihtsalt taustaobjektina. See on kasulik suhtlemiseks objektidega, mis ei pruugi osaleda tavapärastes füüsikalistes interaktsioonides - näiteks seinal olev lüüti.

Kummitus (*Ghost*)

Kummitus annab objektile võime teistest objektidest läbi liikuda. See hoiab objektide füüsika (nt põrkamine, hõõrdumine jne) arvutamisel kokku töötlusaega. Teiste objektidega põrkumine siiski tuvastatakse ja sellest teavitatakse ka mänguloogikat.

Nähtamatu (*Invisible*)

Kui objekt on määratud *nähtamatuks*, siis mängumootor seda ei renderda.

Raadius (*Radius*)

Piirava kera ja materjali füüsika raadius.

Anisotroopne (*Anisotropic*)

Määrab objekti X-, Y- või Z-suunal hõõrdumise suuna. Anisotroopsus on suhteline hõõrdumistakistus mingis kindlas suunas. Võta näiteks rula ratas: edasipidises suunas peaks see lihtsalt pöörlema, kuid külgsuunas liikumine peaks olema keeruline ning seda tuleb kõrge hõõrdetasemega vähendada.

Põrge puudub (*No Collision*)

See lülitab objektide põrkumise välja. Kasuta seda objektide puhul, mis ei saa kunagi dünaamiliste objektidega kokku puutuda, et kiirendada arvutusaega.

Invisible (nähtamatu)

Loe [siit](#)

Dünaamiline (*Dynamic*)

Kui see sisse lülitada, annab ta su objektile reaalsed füüsikalised omadused. Objekti tõmbab nüüd enda poole gravitatsioon, ta pörkub teistelt objektidelt, tal on mass ja teda aeglustab hõõrdejõud.

Seaded

Actor (tegutseja)

Loe [siit](#)

Ghost (kummitus)

Loe [siit](#)

Invisible (nähtamatu)

Loe [siit](#)

Use Material Force Field (kasuta materjali jõuvälja)

Paneb objekti reageerima [dünaamiliste materjalidega](#). Seda ei toeta praegu vaikumisi valitud Bullet Physicsi füüsikateek ning toetab ainult osaliselt praeguse Blenderi versiooni Sumo mootor. Täielikult on ta toetatud Blenderi versioonis 2.24.

Rotate From Normal (pööre normaalist)

Kasuta objekti pööramiseks külje normaali, et see oleks alati pinnast eemale suunatud.

No Sleeping (keela magamine)

Vaikumisi on määratud, et kui veerev objekt jääb seisma, lülitatakse veeremise füüsika välja. Füüsikamootor eeldab, et rohkem ei ole vaja objekti liikumist arvutada. Mõnel juhul vastab see tõele ning seega hoitakse protsessori jõudlust kokku. Teistel puhkudel (nagu jalgpallimängus) tuleb füüsikaarvutusi jätkata ka vahepeal, kuni keegi palli lööb ja selle uuesti liikuma saadab.

Omadused

Mass (mass)

Mass mõjutab objekti liigutamiseks vajamineva jõu hulka. Mida suurem on mass, seda rohkem jõudu tuleb objektile selle liikuma panemiseks rakendada. Mass ei pane objekte kiiremini või aeglasemalt langema. Objekti aeglustamiseks kasuta summutamist ([dampening](#)); kui soovid langemise kiirust suurendada, suurenda [gravitatsiooni](#) (normaalne gravitatsioon on 9.81).

Radius (raadius)

See määrab keraga piiratud objektide piirdkera suuruse ja Fh/FhRot väärtuse. Seda saab suurendada ja vähendada, et anda objektidele soovitud välimus ja käitumine. Piirdala kuju määrab piirde nupp (Bounds), mis on vaikumisi kast (Box).

Piirdala suuruse muutmise ajal näed seda ekraanil ringikujulisena. Praegusel hetkel näidatakse kõiki piirdala kujusid 3D-aknas ringidena, kuigi need mõjuvad mängimise ajal erinevalt.

Form Factor (kujufaktor)

Kujufaktor annab sulle teatava kontrolli jäiga keha objektide üle. Mida kõrgem kujufaktori väärtus on, seda väiksema tõenäosusega hakkab objekt veerema (seda eriti lamedal pinnal).

Anisotropic Friction (anisotroopne hõõrdumine)

Loe [siit](#)

Kiirus (*Velocity*)

Minimum (miinimum)

Piira kiirus antud minimaalse väärtusega.

Maximum (maksimum)

Piira kiirus antud maksimaalse väärtusega.

Summutus (*Damping*)

Translation (nihe)

Summutamine mõjutab objekti liikumisvabadust. Kosmoses ei tohiks summutust üldse olla, samas kui vees peaks summutus olema suhteliselt kõrge.

Rotation (pööre)

Pöörde summutus toimib samal viisil kui liikumise summutus, kuid mõjutab ainult objekti pööret. Kõrge pöörde summutus ei mõjuta objekti sirgjoonelise liikumise momenti (inerti).

Lukusta liikumine ja pööre (*Lock Translation and Rotation*)

Võid objekti pöörde ja liikumise piirata ühe teljega või suvalise X-, Y- ja Z-teljede kombinatsiooniga.

Pörke piirid (*Collision Bounds*)

Loe [siit](#).

Jäigad kehad (*Rigid Bodies*)

Neil on samad sätted nagu [dünaamilistel objektidel](#) (*Dynamic Objects*).

Võimaldavad veeremise füüsikat (*rolling physics*), andes objektidele realistliku käitumise. Kui pall kukub loomulikult tasasele pinnale, oleks imelik, kui see lihtsalt seisma jääks. Veeremisfüüsikaga veereb su pall isegi täiesti tasasele pinnale kukkudes nagu loomulik objekt.

Pehme keha (*Soft Body*)

Füüsikaline objekt, mis pörkel moonutub.

Seaded

Actor (tegutseja)

Loe [siit](#)

Ghost (kummitus)

Loe [siit](#)

Invisible (nähtamatu)

Loe [siit](#)

Omadused

Mass (mass)

Loe [siit](#)

Welding (keevitus)

Welding threshold (keevituse lävi): kaugus naabertippude vahel, mille puhul need loetakse kokkulangevateks => keevituse testi välja lülitamiseks ja stseeni laadimise kiirendamiseks määra väärtuseks 0.0 (lubatud, kui võres ei ole duplikaate).

Position Iterations (asukoha iteratsioonide arv)

Asukoha arvutaja iteratsioonide arv. Täpsema tulemuse jaoks suurenda.

Linear Stiffness (lineaarse jäikuse)

Pehme keha lülide lineaarse jäikuse.

Friction (hõõrdumine)

Dünaamiline hõõrdumine.

Margin (ääre laius)

Pehme keha pörke ääre laius (*collision margin*). Väike väärtus muudab algoritmi ebastabiilseks.

Bending Constraints (painde piirajad)

Lülita painde piirajad sisse.

Shape Match (kuju taastamine)

Lülita sisse pehme keha kuju taastamise sihtmärk, mille määrab läve väärtus (Threshold).

Parvpörked (*Cluster Collision*)

Rigid to Soft Body (jäigad kehad pehmetega)

Lülita sisse parvpörked pehmete ja jäikade kehade vahel.

Soft to Soft Body (pehmed kehad pehmetega)

Lülita sisse parvpörked pehmete kehade vahel.

Iterations (iteratsioonide arv)

Määra parveiteratsioonide arv.

Pörke piirid (*Collision Bounds*)

Loe [siit](#).

Peitja (*Occluder*)

Valikuline peitmine tähendab võimalust, et objektid (peitjad) varjavad teisi objekte (st neid ei saadeta graafikaprotsessorile renderdamiseks). Selle rakendus samaneb puuvalikuga (*tree selection*).

Peitmine ei ole vaikimisi sisse lülitatud, sest sellest kasu saamiseks tuleb temast kõigepealt aru saada. Kui seda õigesti ei kasutata, võib mäng muutuda aeglaseks. (Kuid mitte väga palju, sest arvutamise aega reguleeritakse automaatselt.)

Kuidas see töötab

Peitmine lülitub sisse, kui määratud on vähemalt üks peitja (*occluder*). Peitmise valik asub füüsika nuppude all.

Peitjat tuleks kasutada ainult võrgul põhinevate (*network-based*) objektide puhul. Füüsika enda jaoks on see võrdväärne pörke puudumisega (*No collision*). Põhjus, miks peitmise režiim tühistab kõik teised režiimid, on selles, et peitjad peaks olema spetsiaalselt selleks loodud ja mitte iga võre ei tohiks olla peitja. Kuid sa saad füüsikat kasutataval objektidel peitmise võimaluse sisse lülitada, kasutades Pythonit ja loogikaklotse (*Logic bricks*). Selleni me veel jõuame.

Kui peitjaobjekt siseneb vaatekoonusesse, ehitab mängumootor selle objekti külgedest Z-sügavuspuhvri (*ZDepth buffer*). See, kas küljel on ühe- või kahepoolsed, on väga oluline: Z-sügavuse puhvri ehitamisel kasutatakse ainult ühepoolsete külgede esikülgi ja kahepoolset külgi. Kui vaatekoonuses (*viewfrustum*) on mitu peitjat, liidab mängumootor nad üheks ning säilitab ainult kõige lähemal asuvate külgede andmed.

Z-sügavuspuhvri lahutuse saab määrata maailma seadete (*World*) all oleva nupuga *Occlu Res* (peitja lahutus).

Vaikimisi on lahutus 128 pikslit vaateakna suurima mõõtme jaoks ning teine mõõde määratakse proportsionaalselt. Kuigi 128 on väga madal resolutsioon, on see peitmise (*culling*) arvutamiseks täiesti piisav. Lahutust saab suurendada kuni 1024 pikslini, kuid see nõuab kõvasti protsessorivõimsust.

Mängumootor liigub läbi DBVT puu ja kontrollib iga sõlme puhul, kas peitja varjab seda täielikult, ning kui see on nii, siis seda ei joonistata (samuti kõiki tema sees olevaid objekte).

Et protsessi veelgi optimeerida, ehitab ja kasutab mängumootor Z-sügavuspuhvrit ainult siis, kui vähemalt üks peitja on vaatekoonuses. Kuni seda ei juhtu, ei toimu võrreldes tavapärase vaatekoonusega vaatepiiramisega mitte mingisugust jõudluse vähenemist.

Kuidas neid kasutada

On situatsioone, kus peitjaga vaatepiiramine (*occlusion culling*) ei too mingit kasu:

- Kui peitjad on väikesed ja varjavad vaid väheseid objekte. Sellisel juhul koormavad peitjaga vaatepiirajad lihtsalt su protsessorit.
- Kui piirajad on suured, aga peidavad ainult lihtsaid objekte. Sellisel juhul on targem need objektid graafikaprotsessorisse saata.
- Kui piirajad on suured ja peidavad palju keerulisi objekte, kuid teevad seda etteennustataval moel. Näide: maja täis keerulisi objekte. Kuigi peitjaga vaatepiiramine töötab sellisel juhul hästi, saad parema jõudluse, kui lood spetsiaalse loogika, mis peidab/näitab neid objekte, tehes näiteks objektid nähtavaks ainult siis, kui kaamera siseneb majja.

Peitjaga vaatepiiramisel on enim mõtet siis, kui peitjad on suured objektid (hooned, mäed vms), mis peidavad paljusid keerulisi objekte etteennustamatul moel. Aga ära muretse liiga palju jõudluse üle: isegi siis, kui kasutad seda ebasobilikult, on jõudluse vähenemine piiratud tänu algoritmi struktuurile.

Peitjad võivad olla nähtavad graafilised objektid, kuid pea meeles, et liiga paljude külgedega objektide korral puhul võib Z-sügavuspuhvri loomine olla aeglane. Näiteks maastik ei ole sobilik peitja: sellel on liiga palju külgi ja see kattub liiga palju iseendaga. Peitjad võivad olla nähtamatud objektid, mis on asetatud keerukamate objektide sisse (näiteks keerulise arhitektuuriga hoone "seinte sisse"). Peitjates võivad olla "augud", läbi mille saab objekte näha.

Järgneb võimalik tööskem, kuidas keerulisest maastikust peitjaid luua:

- duplitseeri maastik ja vähenda oluliselt selle külgede arvu
- eemalda kõik horisontaalsed osad ning jäta alles ainult vertikaalse ulatusega osad
- muuda kõik küljed nähtamatuks ning määra nende režiimiks peitja (*Occluder*).

Et seda veelgi optimeerida, võid jagada peitja mitmeks objektiks (tee igast mäest, igast seinast jne eraldi peitja) ning teha sedasama ka maastiku endaga: sedasi ei renderdata neid maastiku osi, mis jäävad peitjate taha.

Seda, kui hästi peitmine töötab, saad kindlaks teha mängu traatvõre režiimis jooksutades.

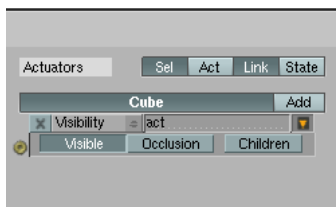
Mängusisene juhtimine

Pythoni abil saad objektide peitjaks olemist sisse ja välja lülitada:

```
obj.occlusion = True
obj.setOcclusion(True, False)    #param1=peitmine, param2=kehtib ka alamobjektidel
```

Objekt ei pea olema kasutajaliidese kaudu peitjana määratletud: peitmist saad iga võreobjekti puhul (staatiliste ja dünaamiliste objektide puhul) mängimise ajal muuta.

Saad kasutada ka nähtavuse käiturit (*Visibility Actuator*):



Pane tähele, et selle käituri paigutus on muutunud: nähtava/nähtamatu (*Visible/Invisible*) nupud, mis enne olid üksteist välistavad, on asendunud kahe eraldi nupuga: vajuta sisselülitamiseks ühte nuppu (*Visible/Occluding*), teine nupp (*Invisible/Non-occluding*) lülitab käituri välja.

Pane tähele, et käitur määrab samaaegselt nii nähtavuse (*Visibility*) kui ka peitmise (*Occlusion*) režiimid. Kui soovid neid eraldi määrata, kasuta Pythonit.

Piirangud

- Kui soovid kasutada peitjaga vaatepiiramist, peab sul mängu alguses olema vähemalt üks peitja. See peitja ei pea asuma aktiivsel kihil. Pärast seda saad peitmist mängu ajal mängusiseste juhtmeetoditega iga võreobjekti jaoks sisse ja välja lülitada.

Sensor

See objekt tuvastab pörked staatiliste ja dünaamiliste objektidega, kuid mitte teiste sensorobjektidega. Sensorobjekt on samane nende füüsika objektidega, millel põhinevad lähedus- (*Near*) ja radarsensorid (*Radar*). Nagu lähedus- ja radarsensorid, on ka sensorobjekt:

- staatiline (*static*) või kummitus (*ghost*)
- vaikumisi nähtamatu (*invisible*)
- alati aktiivne, et pörke tuvastus oleks täpne
- suuteline tuvastama nii staatilisi kui ka dünaamilisi objekte
- ignoreerib pörkeid ülemobjektiga
- võimaline laiaks (broadphase) filtreerimiseks, mis põhineb:
 - tegutsejal (*Actor*): pörkuv objekt peab olema tuvastamiseks tegutseja
 - omadusel/materjalil (*property/material*): vastavalt sellega liidetud pörkesensori määratlusele.

Lai filtreerimine on oluline jõudluse seisukohast: pörkepunktid arvutatakse välja ainult nende objektide puhul, mis läbivad ruumilise filtri.

- eemaldatakse automaatselt simulatsioonist, kui selle küljes ei ole aktiivseid pörkesensoreid

Erinevalt lähi- ja radarobjektidest saab sensor:

- võtta ükskõik millise kuju, kaasa arvatud kolmnurkvõre oma
- muutuda vigade jahtimiseks nähtavaks (kasuta lihtsalt nähtavuse käiturit - *Visible*)
- olla kasutatav mitme pörkesensori poolt

Ülejäänud omaduste poolest on sensorobjektid tavalised objektid. Sa saad neid vabalt liigutada ja hierarhilistesse suhetesse seada. Dünaamilisele objektile allutatuna saavad nad anda neile täiendava kontrolli pörgete üle.

Pörketuvastuse võimalikud tüübid sõltuvad kujust:

- kast, kera, silinder, koonus ja kumer kest võimaldavad ka mahu tuvastamist.
- kolmnurkvõre võimaldab pinna tuvastamist, kuid sa saad pinnale natuke mahtu anda, suurendades täiendavate sätete paneelis (*Advanced Settings*) väärtust *margin* (ääre laius). Ääre laius kehitab mõlema pinna poole puhul.

Nõuanne jõudluse kohta:

- Sensorobjektid on parema jõudlusega kui lähedus- (*Near*) ja radarsensorid (*Radar*): tänu stseenigraafi optimeerimisele teevad nad vähem sünkroniseerimisi ja neil võib küljes olla mitu pörkesensorit (näiteks erinevate omaduste filtritega).
- Eelista võimalusel alati lihtsaid kujusid (kast, kera) keerulistele.
- Kasuta alati laia filtreerimist (väldi pörkesensoreid, mille omadus/materjal on tühi)
- Kasuta pörkesensoreid ainult siis, kui neid vajad. Kui sensorobjekti küljes ei ole aktiivseid pörkesensoreid, eemaldatakse see simulatsioonist ja ta ei kasuta protsessori võimsust.

Piirangud:

- Kui sa käivitad Blenderi veajahtimisrežimis, näed sa konsoolis ühte hoiatusrida:
"warning btCollisionDispatcher::needsCollision: static-static collision!"
Tavalistes Blenderi versioonides seda rida ei näidata.
- Pörke ääre laius (*collision margin*) ei mõjuta kera, koonuse ja silindri kujusid.

Sätted

Invisible (nähtamatu)

Loe [siit](#)

Pörke piirid (*Collision Bounds*)

Loe [siit](#).

Bullet physics Python API

Füüsikateegi Bullet Physics programmiidese abil saab Blenderi mängumootoris kasutada põrketuvastust (*collision detection*) ja jäiga keha dünaamikat (*rigid body dynamics*). See kasutab mõningaid Blenderi seadeid, mis loodi vana põrketuvastussüsteemi jaoks (nimega Sumo).

Aga uutel võimalustel puudub veel kasutajaliides ning seetõttu tuleb senikaua kasutada Pythonit.

Võimalused:

- Masinate simulatsioon.
- Jäiga keha piirajad (*Rigid body constraints*): hingliigend (*hinge*) ja kuulliigend (*ball socket*).
- Ligipääs sisemistele füüsika sätetele nagu deaktiveerimise aeg (*deactivation time*), vigade avastamine (debugging).

Kõige lihtsam viis, kuidas nende kasutust mõista, on vaadata Bullet Physicsi demonstratsioonifaile. Lisainformatsioon saab leida [siit](#).

Pythoni skripti näidis:

```
import PhysicsConstraints
print dir(PhysicsConstraints)
```

Videotekstuuri moodul (*VideoTexture*)

Videotekstuuri mooduliga *VideoTexture* saad mängimise ajal tekstuure muuta. Kasutada saab erinevaid tekstuuride allikaid: videofaile, pildifaile, kaamerast lindistatavat videot, mälupehtrit, Blenderi kaamera renderdust või nende segu. Video- ja pildifaile saab laadida faili nime asemel URLi kasutades otse internetist. Lisaks saad piltidele enne graafikaprotsessorisse saatmist lisada filtreid, mille tulemusena saad luua videoefekte: sinist ekraani (*blue screen*), värviüleminekut (*color band*), halli (*gray*), normaalikaarti (*normal map*). *VideoTexture* kasutab piltide ja videode laadimiseks FFmpeg teeki. Kõike formaate ja koodekeid, mida FFmpeg toetab, toetab ka *VideoTexture*. Sealhulgas:

- AVI
- Ogg
- Xvid
- Theora
- dv1394 kaamera
- video4linuxi lindistuskaart (selle alla käivad paljud veebikaamerad)
- videoForWindowsi lindistuskaart (selle alla käivad paljud veebikaamerad)
- JPG

Kuidas see töötab

Põhimõte on lihtne: esmalt tuvastad sa olemasoleva tekstuuri selle objekti ja nime järgi, seejärel lood sa uue dünaamilise sisuga tekstuuri ja vahetad need kaks tekstuuri omavahel graafikaprotsessoris ära. Mängumootor ei ole vahetusest teadlik ning jätkab objekti kuvamist endistviisi, kuid nüüd on sul täielik kontroll selle tekstuuri üle. Lõpuks uus tekstuur kustutatakse ja taastatakse vana.

Antud lehekülg on mooduli *VideoTexture* juhendiks ning sisaldab lihtsaid näiteid.

Mängu ettevalmistus

Enne mooduli *VideoTexture* kasutamist peavad sul olema õigesti määratud tekstuuridega objekti.d

Kujuta ette, et soovid oma mängu televiisorit, mis näitaks reaalasjas edastatavaid saateid. Lood televiisori objekti ja määrad sellele ekraani kohale (UV-laotusega) mingi tekstuuri, näiteks "tv.png". See, milline see tekstuur välja näeb, ei ole oluline; eeldatavasti soovid sa selle teha tumehalli, et jääks mulje väljalülitatud televiisorist. Kui televiisor sisse lülitatakse, lood sa videolindistuskaardi kaudu dünaamilise tekstuuri ja kasutad seda tv.png asemel: teleriekraan ärkab ellu.

Sul on kaks võimalust määrata tekstuure, mida *VideoTexture* saab haarata:

1. Lihtne UV-tekstuur.
2. Blenderi materjal, millel on pilditekstuuri kanal.

Kuna *VideoTexture* töötab tekstuuri tasemel, on see ühilduv kõigi mängumootori lähedate tekstureerimisvõimalustega: GLSL-i varjutajad, mitmiktekstuurid (*multi-texture*), spetsiaalselt loodud varjutajad jne.

Esimene näide

Oletame, et meil on mängu objekt, mille ühele või enamale küljele on määratud materjal/pilt, millel soovime näidata videot.

Esimese sammuna loome tekstuuri objekti *Texture*. Teeme seda skriptiga, mis käivitub ühe korra. See võib olla mängu alguses; videot ennast mängitakse alles siis, kui sa tekstuuri värskendad - selleni me veel jõuame. Tavaliselt lisatakse see skript selle objekti külge, millel me soovime videot näidata, et objekti viidet (*reference*) oleks võimalikult lihtne leida:

```
import VideoTexture

contr = GameLogic.getCurrentController()
obj = contr.owner

if not hasattr(GameLogic, 'video'):
```

Kontroll, mida me "video" atribuudiga teeme, on lihtne trikk, mille abil saame tagada, et loome tekstuuri ainult ühe korra.

Materjali leidmine

```
matID = VideoTexture.materialID(obj, 'IMvideo.png')
```

VideoTexture.materialID() on kasulik funktsioon, mille abil saab leida selle objekti materjali andmed, mis kasutab tekstuurina pilti video.png. See meetod töötab nii Blenderi materjalide kui ka UV-tekstuuridega. UV-tekstuuri puhul leiab see selle seesmise materjali, mis vastab nendele külgedel, millele see tekstuur on määratud. Blenderi materjali puhul leiab see materjali, millel on pilditekstuuri kanal, mille esimeses kanalis olev pilt on sama kui otsitav nimi.

Eesliide "IM" tähistab, et me otsime tekstuuri nime, kuid märkides eesliiteks "MA" saame me ka materjali otsida. Kui me näiteks tahame leida sellel objekti materjali nimega *VideoMat*, oleks vajalik kood:

```
matID = VideoTexture.materialID(obj, 'MAVideoMat')
```

Tekstuuri loomine

VideoTexture.Texture on klass, mis loob tekstuuriobjekti *Texture*, mis laeb dünaamilise tekstuuri graafikaprotsessorisse. Konstruktor kasutab ühte kohustuslikku ja kolme valikulist parameetrit:

```
gameObj
```

Mängu objekt.

materialID

VideoTexture.materialID() poolt väljastatud materjali indeks, 0 = vaikesi esimene materjal.

textureID

Mitmiktekstuuri kanali (*multi-texture channel*) puhul tekstuuri indeks, 0 = vaikesi esimene kanal.
UV-tekstuuride puhul peaks see parameeter olema alati 0.

textureObj

Viide teisele tekstuuriobjektile `Texture`, mille tekstuuri me soovime taaskasutada.

Kui me seda parameetrit kasutame, ei pea me uue tekstuuri jaoks alustekstuuri looma ning seda pole ka vaja värskendada - see teine tekstuuribjekt `Texture` annab meile mõlema materjali jaoks materjalid/tekstuurid.

```
GameLogic.video = VideoTexture.Texture(obj, matID)
```

Muuda tekstuur püsivaks

Pane tähele, et me oleme selle objekti määranud ekstra selle jaoks loodud mänguloogika `GameLogic` atribuudile "video". Põhjuseks on see, et tekstuuribjekt `Texture` peab püsima erinevate skriptide (või sama skripti mitmete käivituste) vahel. Kohalik muutuja kustutatakse skripti lõpus ning samal hetkel kustutatakse ka graafikaprotsessori tekstuur. Mänguloogika moodulobjekt `GameLogic` on kasulik koht, kus püsivaid objekte säilitada.

Alustekstuuri loomine

Nüüd on meil olemas tekstuuribjekt `Texture`, kuid see ei saa midagi teha, sest tal puudub alustekstuur. Peame looma alusobjekti ühest `VideoTexture` poolt kasutatavast võimalikust allikast:

VideoFFmpeg

Liikuvad pildid.

Videofail, kaamerast lindistatav video, voogvideo.

ImageFFmpeg

Seisvad pildid.

Pildifail, veebis olev pilt.

ImageBuff

Pilt programmi mälust.

Arvuti poolt genereeritud pildid, joonistamisprogrammid.

ImageViewport

Terve vaateaken või selle osa (=aktiivse kaamera poolt ekraanil näidatav renderdus).

ImageRender

Mitteaktiivse kaamera renderdus.

ImageMix

Segu kahest või enamast eelnevast allikast.

Selles näites kasutame allikana lihtsat videofaili. Video konstruktor `VideoFFmpeg` kasutab parameetrina faili nime. Vältimaks faili asukohaga tekkida võivaid segadusi, kasutame absoluutse faili nime ehitamiseks `GameLogic.expandPath()` ja eeldame, et videofail asub samas kataloogis kui `.blend`-fail:

```
movie = GameLogic.expandPath('//trailer_400p.ogg')
GameLogic.video.source = VideoTexture.VideoFFmpeg(movie)
```

Loomes video alusobjekti ja määrame selle tekstuuribjekti `Texture` allika parameetriks `source`, et see oleks püsiv - kuna tekstuuribjekt `Texture` on püsiv, on ka alusobjekt püsiv.

Pea meeles, et me saame tekstuuri `Texture` allikat alati muuta. Oletame, et soovime mängimise ajal kahe video vahel vahetada. Saame teha järgnevat:

```
GameLogic.mySources[0] = VideoTexture.VideoFFmpeg('movie1.avi')
GameLogic.mySources[1] = VideoTexture.VideoFFmpeg('movie2.avi')
```

Ning seejärel saame allikat mängu ajal määrata (ja ümber määrata):

```
GameLogic.video.source = GameLogic.mySources[movieSel]
```

Allika ettevalmistamine

`VideoFFmpeg` allikal on mitu omadust, mis määravad video mahamängimist:

range

[algus,lõpp] (*ujukomaga arv*).

Määra video mahamängimise algus ja lõpp, sekundites faili algusest. Vaikesi mängitakse terve video.

repeat

Korduste arv (*täisarv*).

Video korduste hulk, -1 tähistab lõputut.

framerate

Kaadrisedus (*ujukomaarv*).
Suhteline kaadrisedus, <1.0 aeglasem, >1.0 kiirem.

scale
Skaleerimine (*kahendmuutuja*).
Määra väärtuseks tõene (*True*), et aktiveerida kiire lähima naabri skaleerimisalgoritm.
Tekstuuri laius ja kõrgus peavad olema kahe astmed. Kui videopildi suurus ei ole kahe aste, tuleb mõõtkava teisendada.
Vaikimisi kasutab `VideoTexture` täpset, kuid aeglast `gluScaleImage()` funktsiooni. Kõige targem on video mõõtkava enne ära muuta, et seda ei peaks tegema mahamängimise ajal!

flip
Peegeldus (*kahendmuutuja*).
Määra tõseks (*True*), kui pilt tuleb vertikaalselt tagurpidi pöörata.
FFmpeg mängib videot alati tagurpidi ning seetõttu on selle parameetri väärtus vaikimisi *True*.

filter
Määra videole enne graafikaprotsessorisse saatmist lisafilter.
Väärtuseks määra mõni `VideoTexture` filterobjekt. Vaikimisi saadetakse pilt muutmata kujul graafikaprotsessorisse. Kui videos on olemas alfakanal, laetakse ka see automaatselt ja saadetakse graafikaprotsessorisse.

Me määrame mõõtkava parameetri `scale` väärtuseks *True*, sest `gluScaleImage()` on reaajas mängitava video jaoks lihtsalt liiga aeglane. Kui video mõõtmel on juba kahe astmed, ei mõjuta see säte midagi.

```
GameLogic.video.source.scale = True
```

Video mängimine

Nüüd oleme valmis videot mängima:

```
GameLogic.video.source.play()
```

Video mahamängimine ei ole taustaprotsess, see toimub ainult siis, kui me tekstuuri värskendame. Seega peab meil olema teine skript, mis käivitub igas kaadris, ja käivitab omakorda tekstuuriobjekti `Texture` värskendusmeetodi `refresh()`:

```
if hasattr(GameLogic, 'video'):
    GameLogic.video.refresh(True)
```

Kui video allikas peatatakse, ei oma värskendusmeetod `refresh()` mingit mõju. Värskendusmeetodi `refresh()` parameetriks on kahendmuutuja, mis näitab, kas tekstuuri tuleks värskendamisel uuesti arvutada. Video mängimise puhul pead selle kindlasti määrama tõseks (*True*).

Video staatuse kontrollimine

Video allika klassidel (nagu `VideoFFmpeg`) on staatuse omadus nimega `status`. Kui video mängib, on selle väärtus 2, kui see on seiskunud, on väärtuseks 3. Seega meie näites:

```
if GameLogic.video.source.status == 3:
    #video on peatunud
```

Keerukamad võimalused

Värskendusmeetodi `Texture.refresh()` tõene parameeter (*True*) tühistab lihtsalt pildipuhvri pärast selle graafikaprotsessorisse saatmist, et järgmises kaadris laetaks allikast uus pilt. Selle kõrvalefektiks on see, et Pythoni jaoks on pilt kättesaamatu. Saad seda ka käsitsi teha, käivitades allika värskendusmeetodi `refresh()` otseselt.

Siin on mõned keerukamad võimalused:

- Kasuta pildipuhvrit Pythonis - ei mõjuta tekstuuri (*Texture*):

```
GameLogic.video.refresh(False)
image = GameLogic.video.source.image
# image (pilt) on binaarne stringipuhver peamistest RGBA pikslitest
# ... pildi kasutamine
# tühistab selle järgmise kaadri jaoks
GameLogic.video.source.refresh()
```

- Lae allikast pilt Pythoniga töötlemiseks, laadimata seda graafikaprotsessorisse:

```
# pane tähele, et me isegi ei värskenda tekstuuri (Texture)
# võiksime ka lihtsalt luua alusobjekti ilma tekstuuriobjektita

image = GameLogic.video.source.image
# ... pildi kasutamine
GameLogic.video.source.refresh()
```

- Kui su võrel on rohkem kui üks tekstuurikanal ja sa soovid muuta ühe kindla materjali tekstuuri, leia selle ID

```
matID=VideoTexture.materialID(gameobj, "MAMat.001")
```

GLSL-i materjalidel võib olla rohkem kui üks tekstuurikanal, identifitseeri tekstuur selle pesa (*s/of*) järgi, milles see defineeriti (antud juhul 2)

```
tex=VideoTexture.Texture(gameobj, matID, 2)
```

Lae endale demo

Demo leiad [siit](#).

Keerukamad demod

Siin on [demo](#), mis näitab, kuidas kasutada vaheldumisi kahte videot sama tekstuuri peal. Pane tähele, et see vajab lisaks videofaili, milleks on filmi "Elevantide unenägu" (*Elephants Dream*) reklaamklipp. Demo jooksumiseks võid selle asendada mõne teise failiga.

Siin on [demo](#), mis näitab pildisegamise `ImageMix` kasutamist allikana. `ImageMix` on allikas, mis vajab omakorda allikaid, milleks võivad olla ükskõik milliseid tekstuurile `Texture` sobivad allikad nagu `VideoFFmpeg`, `ImageFFmpeg` või `ImageRender`. Sa määrada nad funktsiooniga `setSource()` ja nende suhtelise kaalu funktsiooniga `setWeight()`. Pea meeles, et kaal on täisarv vahemikus 0 kuni 255 ning kõigi kaalude summa peaks olema 255. `ImageMix` segab kõik allikad vastavalt nende kaalule kokku. Kõigil allikatel peab olema sama suurus (pärast vähendamist lähima kahe astme mõõtmeni). Kui nad seda ei ole, saad sa konsooliaknas Pythoni veateate.

Võimalused

Lühike ja poolik nimekiri mängumootori võimalustest:

- Platvormid: GNU/Linux, Windows, Mac...
- Loogikaklotside (*logicbrick*) süsteem.
- Pythoniga skriptimine.
- Toetab GLSL-i varjutajat 2.0 - tippude (*vertex*) ja fragmentide (*fragment/pixel*) programmid läbi Pythoni või sisseehitatult.
 - Normaalkaart (*normalmap*).
 - Parallaksi kaart (*parallaxmap*).
 - Läikekaart (*speculamap*).
 - Värvikaart (*colormap*).
 - Detailkaart (*detailmap*).
 - Keskkonna peegeldus - kuup (*environment map cube*).
 - Keskkonna peegeldus - kera (*environment map sphere*).
 - Tipuvärvid (*vertex colors*).
 - Phongi varjutaja.
 - Dünaamilised varjud (pehmeid ja šabloonvarje (*stencil shadows*) veel ei ole).
 - GL-i osakeste süsteem (veel ei ole).
- Füüsika: *Bulleti* füüsikamootor ja Sumo.
- Heli: wav (ogg, mp3 veel ei ole).
- Tekstuurid: tga+alpha, png+alpha, jpg.
- Rasterfondid (*bitmap fonts*).
- Graafiline kasutajaliides (veel ei ole).
- Võre deformeerimine luudega.
- Maastikumootor (veel ei ole).
- Detailsuse tasemed (*level of detail*) - Pythoni abil.
- Vaate piiramise süsteem (*culling*): kera, kast, koonus (Pythoni abil).
- Portaalid (veel ei ole).
- Võrgu tugi (Pythoni kaudu).
- Mitu UV-koordinaatide kihti (Pythoni kaudu).
- Mipmap.
- Valgus- ja varjukaartide eelrenderdus.

Välised teabeallikad

Väline dokumentatsioon

- [Gamekit 2.0](#)
- [Algne Blenderi GameKit](#)
- [Materjali GLSL-i varjutajate kasutamine](#)
- [Välised õppetükid](#)

Kogukond

- [Blenderartists](#) Mängumootori tugifoorum ja arutlused
- [#gameblender](#) IRC vestluskanal

Arendus

- [Mängumootori arendus](#) (wiki lehed)
- [Mängumootori veajahtija](#)

Mängude arendamise tööriistad ja vahendid

- [Pilditöötlusprogramm GIMP \("GNU Image Manipulation Program"\)](#) GNU/Linux ja Windowsi jaoks
- [IconArt](#) - tee oma game.exe jaoks ise ikoonid
- [Setup2go and Installer2go](#) - installeerija loomine
- [Inno Setup](#) - väga hea tasuta tarkvarapaigaldaja (installer)
- [BZoo](#) - väga võimas võrgumängu mall
- [Mängumootori võistlus](#)
- [WSAG](#) - Korralik mitme mängijaga võrgusüsteem

Python

- [Pythoni märkmed](#)
- [Mängumootori Python API dokumendid \(2.49\)](#)